METHODS FOR CONSTRUCTION OF SURROGATES FOR COMPUTATIONALLY EXPENSIVE HIGH-DIMENSIONAL PROBLEMS

A Thesis
Presented to
The Academic Faculty

by

Dushhyanth Rajaram

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in
Aerospace Engineering

School of Aerospace Engineering
Georgia Institute of Technology
December 2020

Copyright © 2020 by Dushhyanth Rajaram

METHODS FOR CONSTRUCTION OF SURROGATES FOR COMPUTATIONALLY EXPENSIVE HIGH-DIMENSIONAL PROBLEMS

Approved by:

Professor Dimitri Mavris, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Professor Jechiel Jagoda
School of Aerospace Engineering
Georgia Institute of Technology

Professor Ümit V. Çatalyürek
School of Computational Science and Engineering
Georgia Institute of Technology

Dr. Olivia Pinon Fischer
Senior Research Engineer
Georgia Institute of Technology

Dr. Frederic Villeneuve
Principal Systems Lead
Blue Origin

Date Approved: October 13th 2020

To my parents...

ACKNOWLEDGEMENTS

The process of developing and writing this dissertation has been a humbling and fulfilling experience. Achieving this ultimate academic degree would not have been possible without the unequivocal support, constant encouragement, and care of several individuals throughout this long, and sometimes seemingly endless journey. It goes without saying that this achievement is as much theirs, as it is mine. I would like to express my appreciation and acknowledge my debt towards all of them.

I would first like to start by expressing my utmost appreciation and gratitude towards my advisor and guru – Professor Dimitri Mavris, whose teachings and guidance in matters related to both academics and life in general, have been instrumental in shaping and honing my outlook and perspective in life. Despite his busy schedule, I cannot count even one instance over the course of six long years, when he was not available to provide advice, much needed encouragement, and inspiration to keep the energy, spirit, and humor running high all the time. It is extremely difficult to express how grateful I am to Doc for all he has done and continues to do for me.

I would also like to express my appreciation for all the members of my defense reading committee: Professor Jechiel Jagoda for his insightful questions, candidness and humor; Professor Umit Catalyurek for agreeing to participate at the last minute and providing some insightful comments regarding issues related to computational aspects; Dr. Olivia Pinon Fischer for assiduously reading the rather boring dissertation cover-to-cover, catching errors that missed my eyes, and providing detailed and insightful recommendations for improvement; Dr. Frederic Villeneuve for always reminding me to keep the broad picture in mind, giving me valuable practical advice regarding professional matters, and always highlighting my strengths. I must specially thank Dr. Olivia Fischer for believing in my abilities and standing up for me whenever required.

It would be remiss of me to not acknowledge and express my gratitude towards my

colleagues and friends, for without the relationships I had the pleasure of cultivating with them, this arduous journey would not have been enjoyable (perhaps possible even). While everyone cannot possibly be named, I would like to say thanks in no particular order to a few individuals: Devan, Kiran, Ollie, and Eswar for sharing a house with me and being affectionate despite my mostly annoying habits; Raph, Coco, Sukanya, Sanju, Tapan, Lalla, Vivek, Arturo, Garima for the great bond we have developed and continue to foster; Mayank, Ameya for being close buddies and compatriots; Darshan, Ruxi, Lea for the light-hearted banter and fun; Melek for being a good friend and a co-TA; Christian for his willingness to share his CFD expertise and co-author papers with me; Raph for long discussions about various topics (sometimes random things) and weirdly growing fond of each others' annoying habits; Ramana and Satyam for discussions on everything under the sun. I must specially recognize Tejas for the friendship we have developed, our long discussions on technical topics and strategies and plans for professional growth. I must also thank Manish for our long random coffee walks where we shared the ups and downs of pursuing the degree and discussed about various things. I believe that our rather childish whim to maintain identical timelines for proposal and defense somehow pushed both of us to graduate earlier rather than later. Finally, I must express my profound gratitude to Tanmay Rajpurohit, whose willingness to impart context independent, pure knowledge has equipped me with the perspective to see through situations clearly.

I would also like to recognize the individuals that I consider as family. A very special and deep appreciation and gratitude for Asmita, whose constant support, unwavering love and affection through the rather dramatic ups, and more importantly, downs have helped me and will continue to help me for years to come. Words cannot express her positive influence on my life. Her kind heart, infectious smile, and positive outlook and hope have the potential to move mountains. Heartfelt yet most understated thanks to Siddart, who has grown to be more like a brother, and has influenced me in ways I cannot explain. Even distance and time do not seem to affect the strong bond and understanding of each other we share. Thanks to Karthik, one of my closest and oldest friends with whom I share an irreplaceable bond, for constantly and patiently encouraging me. A special thanks to Yuri,

Aditi, Mams, Saha, Negi, Sanks, and Pattu for their appreciation and encouragement.

No words can do justice to the endless sacrifices my mother – Kanthimathi Rajaram, and my father – Rajaram Krishnan, have made, and continue to make for me. Since my childhood and even today, they enthusiastically encourage me to pursue whatever I wish to, even though at times, my choices are arguably questionable. Their incessant determination is the singular reason I can afford to stand here today. Neither actions nor words can ever equal or express what my parents have done for me.

Dushhyanth Rajaram

Atlanta, GA

November, 2020

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Maturation of computational models through the past several decades has increased reliance on numerical simulations for the analysis, and more importantly, design of complex engineered systems. This paradigm shift can be evidenced by emergence of aerospace startups such as *Boom* and *Aerion* in a rather capital intensive industry demanded by the erstwhile dependence on physical experiments. The high accuracy and realism offered by simulation-based analysis and design often comes at a high computational cost especially in the *many-query* context, as such limiting its applicability in exploratory design studies. In such situations, it is commonplace to rely on models that exploit simplifying assumptions to circumvent the issue of high computational cost. In scenarios where such assumptions do not hold, practitioners often resort to computationally cheap surrogate-based methods. Based on the nature of outputs, the literature often distinguishes between methods that approximate spatio-temporal discrete field quantities and ones that substitute a scalar function of the field.

The primary goal of this dissertation is to improve methods that rapidly approximate spatially distributed high-dimensional outputs with a special emphasis on their applicability to practical problems. The work specifically focuses on the following issues related to the construction of surrogate models that approximate field outputs: 1) the need to handle large datasets under constrained computational resources, 2) the ability to handle a large number of inputs, and 3) the trade-off between computational cost and the need for high-accuracy when the underlying analysis is expensive and has a large number of input parameters.

Pursuit of the first issue investigates the viability of randomized algorithms as a means to perform computationally efficient data compression while preserving accuracy. Three randomized dimension reduction algorithms' applicability and effectiveness for constructing

POD and interpolation-based predictive surrogate models are empirically analyzed. Findings show that randomization is an efficient and sufficiently accurate strategy to compress large datasets and yield surrogate models that are competitive in their predictive accuracy.

Our attempt to address the second issue i.e. accommodate a large number of inputs leads to the formulation of a manifold optimization-based Gaussian Process regression model that simultaneously finds a low-dimensional input subspace and trains a model in it using input-output pairs exclusively. The proposed model is shown to be competitive with contemporary methods that are designed to work under similar constraints. To enable construction of predictive surrogate models that emulate field outputs for analyses with a large number of inputs, we propose the use of the developed manifold optimization-based model to learn a map between each POD coordinate and the inputs by discovering and training a surrogate model in a low-dimensional input subspace. The developed methodology allows construction of accurate predictive field surrogate models for truly practical problems with high-dimensional inputs and outputs using a modified POD and interpolation-based method.

As a potential solution to the third issue regarding insufficiency of expensive simulation data for achieving a desired model accuracy, this dissertation proposes the use of data from multiple sources varying in fidelity. The solution utilizes a recently proposed manifold alignment-based technique to obtain compressed representations of field solutions from disparate analyses. To learn the map between the resulting multifidelity compressed dataset with a large number of inputs, we propose and formulate a multifidelity extension to the manifold optimization-based Gaussian process regression model. The model leverages abundance of low-fidelity data in addition to some high-fidelity data to improve the quality of a common low-dimensional input subspace that captures majority of the variation of the compressed representation obtained from a multifidelity field solution dataset.

For each focus area, the effectiveness of our proposals is demonstrated on field output emulation tasks for practical application problems. The results suggest that the techniques investigated in this dissertation have the potential to enable construction of surrogate models for truly large and practical problems typically encountered in engineering design.

CHAPTER I

MOTIVATION

1.1 Introduction

The past several decades have witnessed steady, rapid strides in the ability to perform numerical computations to simulate the behavior of complex engineered systems. These advances have led to a proliferation of tools that accurately simulate physical governing laws prescribed by nature. They equip engineers with machinery to deal with complex natural phenomena under which engineered systems must function.

Despite the enormous capabilities afforded by computational tools, conceptual engineering design is largely based on algebraic equations that are valid under simplifying assumptions. The models that result enable cheap and fast calculations revealing important high-level trades, making them highly desirable and convenient. For instance, in the realm of aerospace engineering, the design and optimization of conventional tube-and-wing configurations [28, 55, 202] has relied on historical data [212, 218] and simplified equations [122] to compute preliminary estimates of thrust, weight, and wing area in early/conceptual design phases. The success of these methods may be attributed to the fact that vehicles grouped together based on their purpose predominantly looked identical, were manufactured using conventional materials, had comparable aerodynamic performance, and employed similar power plants. The designs had largely common couplings between the various disciplines, the only major difference being size and scale.

Aggressive environmental goals put forth by NASA's [71, 217] "$N+i$" programs have spurred organizations to develop new concepts in a bid to meet demands/projections in the decades to come. Consequently, novel designs featuring several technologies related to advanced materials (e.g. composites), efficient propulsion and aerodynamics have emerged, as shown in figure 1. One of the common features (and design challenge) among all the

a) X-57 Maxwell (credit:NASA)

b) N3-X (credit:NASA)

c) Truss-based Wing (credit:Boeing)

d) QueSST (credit: Lockheed Martin)

Figure 1: Advanced Fixed-Wing Concepts

unconventional designs is a relatively stronger interplay among the aerodynamics, propulsion, and structures disciplines. It is hoped that through the effects of these couplings, favorable performance to meet design goals can be achieved. To account for the departure from conventional designs, traditional techniques (mentioned above) may be adapted using the so-called *k-factor* approach. *k-factors* are multiplicative/additive factors that approximately account for new physics in the equations for known designs. In conjunction with data and a calibration procedure, these methods are applicable (even today) for performing rough, rapid, back-of-the-envelope calculations. It must be noted that numerical simulations and/or experiments serve as enablers for the *k-factor* approach. In general, applicability of the *k-factor* approach is limited within the range of available data because data serves as the truth in the calibration procedure.

Unsurprisingly, practitioners have therefore started recognizing the need to incorporate

detailed models [17, 106, 227, 239, 261, 276] that are relatively more realistic in the assumptions they make, early on in the design process. Resulting models require solving governing laws (e.g. Navier-Stokes, Potential Flow etc.) that are naturally posed as differential equations. Unfortunately, a majority of the practically applicable governing laws do not have closed-form solutions and must be solved using spatial and temporal discretization leading to a large number of unknowns. In essence, they solve for a finite dimensional approximation of a field varying in space (and time). However, the realism and accuracy come at the cost of higher computational complexity limiting their value for use in rapid trade-studies and design space exploration (DSE) exercises desired in early/conceptual stages.

In acknowledgement of the need for early adoption of analyses involving computationally expensive, high-fidelity numerical simulations and/or experiments, practitioners often resort to surrogate models that serve as cheap alternatives. Roughly speaking, surrogate models mimic the behavior of an underlying experiment or numerical simulation using an approximate function, thereby enabling rapid predictions at the cost of some loss in accuracy. Creating them involves a training procedure requiring data from the *true* process they are built to approximate. The training procedure is posed as an optimization problem designed to minimize the error incurred in prediction using the surrogate. Furthermore, one among many classifications implicitly distinguishes between surrogates [1] that are used for predicting scalar quantities of interest and ones that are used for predicting field quantities resulting from solutions of spatially and/or temporally discretized PDEs. Methods belonging to the latter are commonly called Reduced Order Models (ROMs). Before proceeding, let us pause to highlight the high-level needs that motivate this work in light of the discussion so far: 1) The need for incorporation of higher-fidelity models in early design stages; 2) Owing to the computational cost associated with accurate analyses especially involving discretized PDEs, the need to create ROMs that enable rapid trade-studies and DSE exercises.

[1] The term surrogate typically refers to methods that approximate scalar functions. In this work surrogates are used in a much wider sense to include methods that approximate an underlying computationally expensive process. We intend for its specific meaning to be clear depending on the context.

Chapter Outline. This chapter lays the foundation for formulating the main motivating high-level research question that this work attempts to investigate. The role of numerical simulations in the paradigm-shift to simulation-driven design is first introduced followed by the many-query context and a historical perspective on the use of surrogates in such contexts along with the broad classifications of surrogates that exist in literature. Finally, we summarize the discussions with some observations leading to the motivating research question.

1.2 The Paradigm-shift to Simulation-driven Design and Engineering

In the absence of highly accurate models of physical phenomena and/or the inability to simulate them, engineers have depended on experiments to predict the behavior of engineered systems. Literature on this abounds with examples. In the aerospace engineering community especially, it is not uncommon to build a scaled prototype of an aircraft/wing and perform performance tests on it in the wind tunnel. Time and resources demands of such experiment-based processes result in large design-cycle times, more so when a change must be made entailing re-building and re-testing. Even in cases where a detailed model is available, experiments have served as the source of final validation.

Arguably, the major bottleneck has been the lack of computational power. As mentioned earlier however, massive leaps have been made in the development of high-fidelity mathematical models, efficient numerical algorithms, and approximation techniques for many engineering applications. Computing power has steadily risen in agreement with Moore's law [228] leading to record breaking simulations of practical interest [177]. A classic example supporting the claim is worth mentioning here. A highly accurate algorithm called Direct Numerical Simulation (DNS) was proposed by Orzag et. al. [185] in 1972. However, due to lack of compute power, it went on to be implemented more than a decade later by Rogallo et. al. [216]. Furthermore, researchers [270] have sought allegiance in simulations using DNS in situations where experiments were either too challenging or impossible. Such developments, in addition to the need for high-fidelity tools (discussed in section 1.1) indicate a marked paradigm shift, one that has slowly started recognizing and accepting numerical simulations

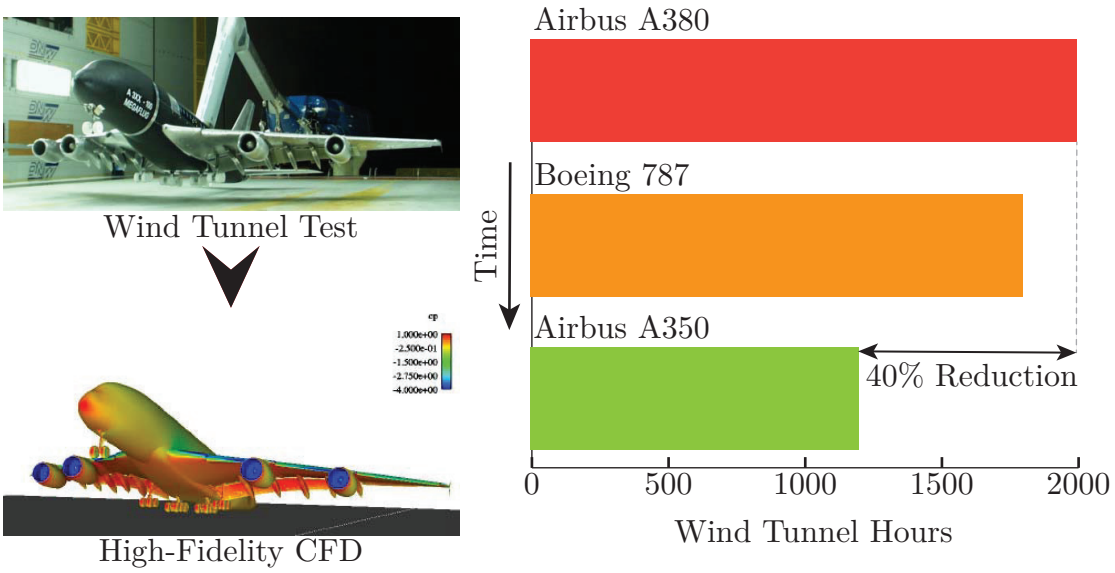


Figure 2: Decline in Wind Tunnel Use Over the Years

as a substitute (wherever appropriate) for physical experiments.

Yet another source of evidence for the shift to simulation-driven design is the emergence of aerospace startups like *Boom Aerospace*, *Spike Aerospace*, and *Aerion Supersonic*. In a capital intensive industry dominated by a handful of competitors, these ventures are now possible only because of trustworthy numerical simulations [5]. As a result of the ability to perform faithful numerical simulations that can replace expensive physical experiments, small companies are able to defer physical experiments to later stages of design, thereby decreasing the financial burden. Lastly, industry dominating companies such as Airbus and Boeing have openly adopted this paradigm-shift. For example, the Boeing 777 aircraft was completely pre-assembled and designed digitally i.e. on computers [103]. In fact, as shown in figure 2, the Airbus group has published some data indicating a significant decline in the number of wind tunnel hours associated with aircraft design campaigns over the years. An open report [93] plainly attributes a reduction in design-cycle time to high-fidelity CFD simulations. Undeniably, we are in the midst of a transition to the era of simulation-driven design enabled by the rise in compute power and algorithmic advancements. For an extensive discussion with examples, the reader is encouraged to read recent work by Karlberg et. al. [126].

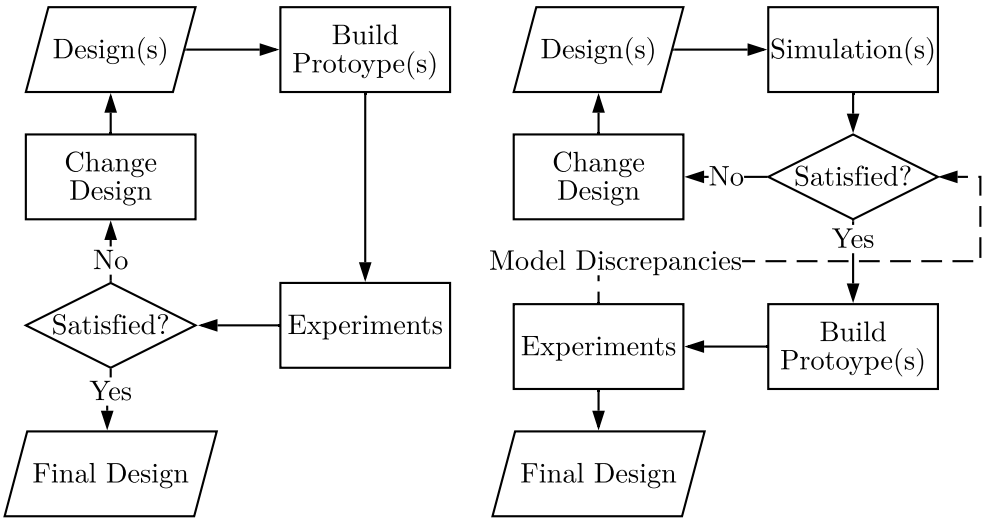


Figure 3: Shift to Simulation-Driven Design

While reliance on numerical simulations in the design process is on the rise, physical experiments/tests still retain the privilege of having the final say. As shown in figure 3, simulations merely circumvent the loss of time and resources by replacing and/or delaying the onset of physical experiments early in the design process. However, because they offer a relatively higher realism and accuracy, most of the critical design changes can be made well in advance. Moreover, effects of very complex natural phenomena can be accounted for because they can be predicted. Note that irrespective of the approach, design is an iterative process. In fact, some well-known practitioners [167] argue that design is naturally posed as an optimization problem which is an iterative process by definition. Therefore, multiple evaluations of a model (physical or digital) embedded in a loop characterizes the design process. In such iterative processes, presence of high-fidelity analyses presents its own set of unique challenges. The biggest challenge is the computational cost associated with high-fidelity simulations. As is often the case, evaluation of even a single configuration of a moderately complex system can take time in the order of days to yield results on supercomputers because solving governing laws is inherently a computationally expensive endeavor. Moreover, it is not hard to imagine situations where multiple configurations/scenarios/designs must

be evaluated until a set of criteria is satisfied. Put together, these challenges make the adoption of high-fidelity numerical simulations unwieldy and intractable in most realistic cases involving iterations.

1.2.1 Outer-loop Applications and Many-query Context

The terms *many-query* [221], *real-time* [221], and *outer-loop* [192] have been used in the literature to formalize applications that require evaluation of a model multiple times. Although certain subtle differences exist in their definitions, for purposes of this work they all bear the same meaning. Loosely speaking, outer-loop is used where a notion of convergence terminates the iteration. For example, in each iteration of an optimization exercise, the algorithm provides inputs (design variables) to the model for evaluation of the objective, constraints and their gradients (if available). The algorithm methodically hones in on the optimum and terminates upon success as specified by a criterion. Uncertainty quantification (UQ) and propagation are examples of outer-loop applications where either an unknown distribution must be inferred or some statistics of an underlying distribution must be estimated. Convergence is determined by the quality of entities being computed. The term many-query, on the other hand refers to a broader set of applications where a particular outer-loop result is of no concern. Examples include parameter studies, DSE, and trade-studies.

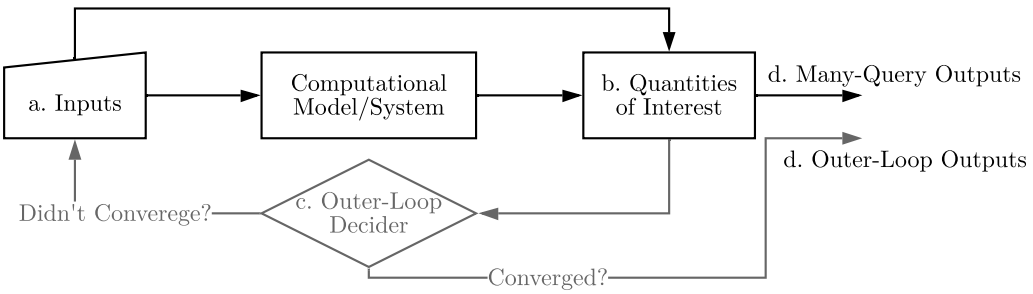


Figure 4: Many-Query and Outer-Loop Contexts

It must be remarked that the process of designing a complex engineered system involves heavy use of situations (detailed in table 1) that fall under the many-query context (shown in figure 4). Since analyses that demand prohibitive computational costs must be incorporated in such applications, let us now take a brief look at some techniques that practitioners have

relied on for respite. In order to precisely motivate the proposed work, emphasis will be laid on emerging needs, shortcomings, and challenges associated with each approach.

1.3 Promise of Surrogate Models

Many techniques have been used to circumvent intractability attributed to high computational costs in many-query applications. One of the most popular and common methods is the so-called surrogate modeling approach. Literature on the use of surrogates dates back to papers published in the 1970s [176] followed by multiple other early studies [26, 237]. Roughly speaking, surrogates are mathematical constructions intended to substitute an expensive model. Among several noteworthy salient features is their relatively inexpensive computational evaluation cost. Additionally and most importantly, their construction process ensures minimization of the loss in accuracy as compared to the expensive model. In mathematical terms, say the model of a physical phenomenon is denoted by an expensive function $f : \boldsymbol{\mu} \mapsto \mathbf{y}$, mapping the variables $\boldsymbol{\mu} \in \mathbb{R}^{n_\mu}$ to $\mathbf{y} \in \mathbb{R}^{n_y}$. The process of surrogate modeling creates an abstract object denoted by the function $f' : \boldsymbol{\mu} \mapsto \mathbf{y}'$, $\mathbf{y}' \in \mathbb{R}^{n_y}$ such that $\mathbf{y}' \approxeq \mathbf{y}$. Although the cost of evaluating the surrogate model is insignificant compared to the model it mimics, the process of creating it may be relatively expensive. The literature often uses the terms *offline* and *online* to disambiguate the benefits of using surrogates. While *offline* cost refers to the often large up-front cost to create the surrogate, *online* cost refers to the inexpensive evaluation of the approximate model in the many-query context.

Table 1: Examples of Many-Query and Outer-Loop Applications (maps into figure 4)

	DSE /Trade-Studies	**Optimization**	**UQ /Propagation**
a.	Design/Decision Variables	Design/Decision Variables	Design and/or Random Variables
b.	Scalar Functions of State and/or Design Variables	Scalar Functions of State and/or Design Variables	Scalar Functions of Random , Design, and State Variables
c.	NA	Optimization Algorithm	Application Specific Algorithm (Eg. Sampling)
d.	Visualizations/Summaries /Dashboards	Optimum Design(s)	Probabilities, Response Distributions, Moments

It is constructive to view surrogate modeling methods as means of producing commodities that once created, can be used in a variety of applications despite the large up-front cost. In this sense, incurring the large upfront cost is justified. This perspective elucidates the benefits of using surrogates in lieu of computationally expensive functions.

Among other classifications in the literature, Eldred et. al. [88] organize surrogate-based methods into the following: *data-fit* models, *projection-based ROMs*, and *hierarchical* models. As their name suggests, *data-fit* models are essentially response surface methods [178] employing either interpolation or regression on a data set of scalar-valued functions and the corresponding input/design/decision variables/parameters obtained from a simulation/physical experiment. On the other hand, *hierarchical* surrogates exploit model hierarchies among multiple levels of fidelity (such as coarse grid simulations, loose solver tolerances, physics-based simplifications) and accompanying data to derive the surrogate model. These surrogate models are often referred to as multifidelity, variable fidelity/complexity models. They find extensive use in optimization-based applications where rigorous mathematical frameworks [6, 102, 215] guarantee global convergence to the highest-fidelity model. Lastly, *projection-based ROMs* decrease the number of unknowns in the discretized governing equation set by exploiting data compression techniques from linear algebra.

As discussed, the use of surrogates as a means to enable many-query applications with expensive analyses is ubiquitous in the engineering literature. However, the choice of a specific class of methods in practice is dictated by the requirements/constraints and characteristics of the application. Let us now examine two different perspectives that exist in the literature, but are not explicitly discussed, with regard to approximation of field quantities versus approximation of scalar functions. In particular, let us focus on its ties with machine learning, discuss differences and note observations to motivate the main objective of this work.

1.3.1 Approximation of Scalar Functions

In practice, engineers are often concerned with the behavior of a scalar functions (quantities of interest or QOIs, in short) of design/decision variables defined over an appropriate

domain. These scalar functions are typically expressed in one of the following ways: 1) analytical equations in which design variables assume the role of independent variables and 2) integrated quantities which are functions of the design variables and high-dimensional fields (commonly known as states) arising from the solution of a discretized set of differential equations (eg. lift, drag, mean temperature).

Surrogate models that directly approximate the set of scalar QOIs by bypassing the underlying governing law (for the latter type of scalar functions) fall under the category of *data-fit* models. In other words, natural mechanism of the underlying physical law is completely ignored when creating these kinds of models. Purely reliant on a set of input-output pairs, they treat the simulation/experiment as a black-box. Their success rests on the idea/assumption that the phenomena encapsulated in a black-box data can be captured exclusively by data in the form of input-output pairs.

The process of creating such surrogates entails creation of a sampling plan in the decision/design variable space, popularly called Design of Experiments (DOE). In a nutshell, the basic indicators of a good DOE are parsimony in the number of samples, good coverage of the domain, and high density of points in regions where high nonlinearity is expected. Albeit under different names, the subject of DOEs has been discussed extensively in the literature [225, 226, 233, 237] with emphasis on engineering applications. Once the DOE is created, the function(s) to be emulated is/are evaluated to obtain input-output pairs. This is arguably one of the most expensive steps followed by a regression or interpolation of the data to a fixed functional (in many cases) form. Both these steps together constitute the *offline* cost. The choice of a particular functional form of the surrogate model has been a subject of discussion and evolution. Early applications saw heavy use of polynomial response surfaces [92, 178]. Although simple in their formulation, they are still relevant and widely used. Their only major drawback is poor performance in emulating highly nonlinear and multimodal functions [26]. More advanced techniques like Radial Basis Functions (RBFs) [48], Gaussian Process Regression (GPR) [209] (a popular variant of which goes by the name

of Kriging [134, 170, 225]), [2], and Artificial Neural Networks (ANNs) [78, 200, 260, 281] followed suit. It is worth mentioning that GPR takes a slightly different approach in that the observed function values are modeled as a collection of random functions, a finite number of which follows a Gaussian distribution. The relationship between data-points is modeled using a distance-based covariance function. An attractive consequence of its mathematical formulation is the availability of error (uncertainty) in prediction. Recent application of deep variants of the GPR [205] for engineering problems has shown immense promise. The cost of evaluating the approximation/surrogate or the *online* cost for most scalar surrogates is negligible with the exception of GPR/Kriging. For Kriging and GPR, the computational cost for updates to the prediction formula along with the requirement to compute pair-wise distances scales strongly with the total number of data points.

This general perspective towards surrogate modeling may be considered mature because it has evolved over many decades as evidenced by several review and survey articles. Comprehensive studies include the ones by Queipo et. al. [199], Forrester et. al. [95], Yondo at. al. [272], and Bhosekar et. al. [38] among many others [96, 233, 234, 258]. Before proceeding, it is instructive to mention a few noteworthy points:

1. Surrogates for scalars completely bypass the underlying governing law (whenever present).

2. They are purely data-based methods and have had considerable success in practice. They are also non-intrusive in the sense that they only require input-output pairs.

3. Because they ignore the field solutions from the governing law, they are of no value in situations where emulation of the field output is of clear interest.

1.3.2 Approximation of Discretized Functions or Fields

Most intricate and complicated natural phenomena are elegantly modeled as partial differential equations (PDEs). They express an unknown quantity (function) that varies in

[2] The term *Krigeage* was introduced by Matheron [170] in honor of engineer Danie Krige [134] who invented the method. Sacks et. al. [225] led its usage to the engineering design domain where it assumed the name of Kriging.

space (x, y, z) and/or time (t) following certain rules dictated by spatial and/or temporal derivatives. Their complexity manifests itself through our inability to obtain *closed-form* solutions. Therefore, numerical algorithms solve for a discrete approximation of the unknown function by discretizing the equation in space and/or time. As one can imagine, this process results in a set of equations with a large set of unknowns. The resulting high-dimensional solution is commonly referred to as a *field*. [3]

In practical cases, the size of unknowns can easily reach $\mathcal{O}[10^6 - 10^8]$. Examples in engineering include fluid dynamics, structural dynamics, heat transfer, chemically reacting flows etc. In the many-query context, performing many simulations for different values of decision variables in large-scale settings places prohibitively large demand on computational resources. A solution to alleviate this computational burden is the main role of Reduced Order Models [31]. This specific type of surrogate modeling falls under the class of *projection-based ROMs* as defined by Eldred et. al. [88]. A broad-view of ROMs proposed by Antoulas [16] is shown in figure 5.

Instead of approximating scalars, ROMs are designed to rapidly emulate high-dimensional fields like those arising from the solutions of discretized PDEs/ODEs. ROMs operate under the assumption that most solutions can be approximated in a subspace whose dimension is much smaller than that of the original system. Reduction is achieved by projecting the discretized equations onto an appropriate low-dimensional subspace (more generally called a manifold). Additionally, the ability of ROMs to retain the underlying mathematical structure of the system is touted as what makes them *physics-based* when compared to *data-fit* approaches. On further examination, one can argue that this claim is not always true. Nevertheless, they are extremely useful in circumstances when distributed field quantities must be passed between coupled analyses, as shown in figure 6. A common use case is the natural coupling between the fluid forces on the wing and deflection (i.e. shape). Physically speaking, both the aerodynamic forces and structural deflections must be consistent, i.e. the forces generated due to the shape of the wing under deflection must give rise to the deflections that generated those forces leading to an organic iterative procedure where fields must

[3] Consistent with the usage in physics. Not to be confused with the definition of *field* in mathematics

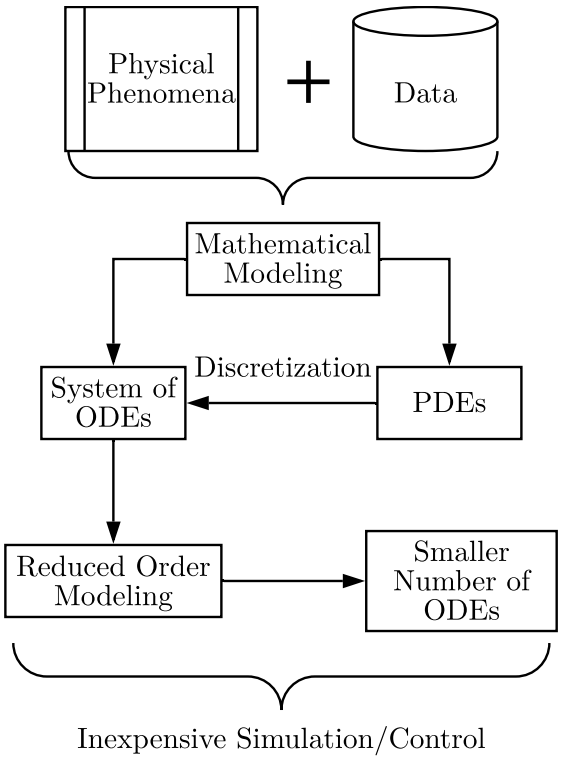


Figure 5: A High-Level View of Reduced Order Modeling [16]

be passed between analyses. Even taken one at a time, for practical cases, solvers for the structural and aerodynamic analyses can be computationally expensive. Instead of using the solvers themselves, one can deploy ROMs to quickly predict the force and/or deflection fields thereby speeding up convergence. One must note that in such cases, surrogates for scalar functions, as defined in section 1.3.1 are clearly inadequate in providing the required information. One may argue that a scalar surrogate can be trained for the field value at each grid point. Although technically viable, such an approach falls short because of the following reasons: 1) computational inefficiency (high memory requirements) because of the requirement to train surrogates equal in number to the high-dimensional field's dimension, 2) lack of spatial and temporal coherence considerations that may exist within fields, and 3) the challenge of handling dynamic field evolution. There is yet another reason that makes ROMs advantageous. If the approximation of a scalar function is indeed the goal, fields emulated by the ROMs along with the design variable values can be used to inexpensively compute the scalar objective. For instance, the process of drag and lift computation for the

flow over a body clarifies this argument. Say a ROM predicts the pressure and shear stress field on a lifting surface. The knowledge of geometry (say the design variables, in this case) and the field can be used to compute the drag (a scalar objective). This example shows how a ROM is also useful for predicting scalar functions. Note that this is possible exclusively in cases where evaluation of the model consists of an intermediate step where a system of discretized equations is solved in order to respect a governing law.

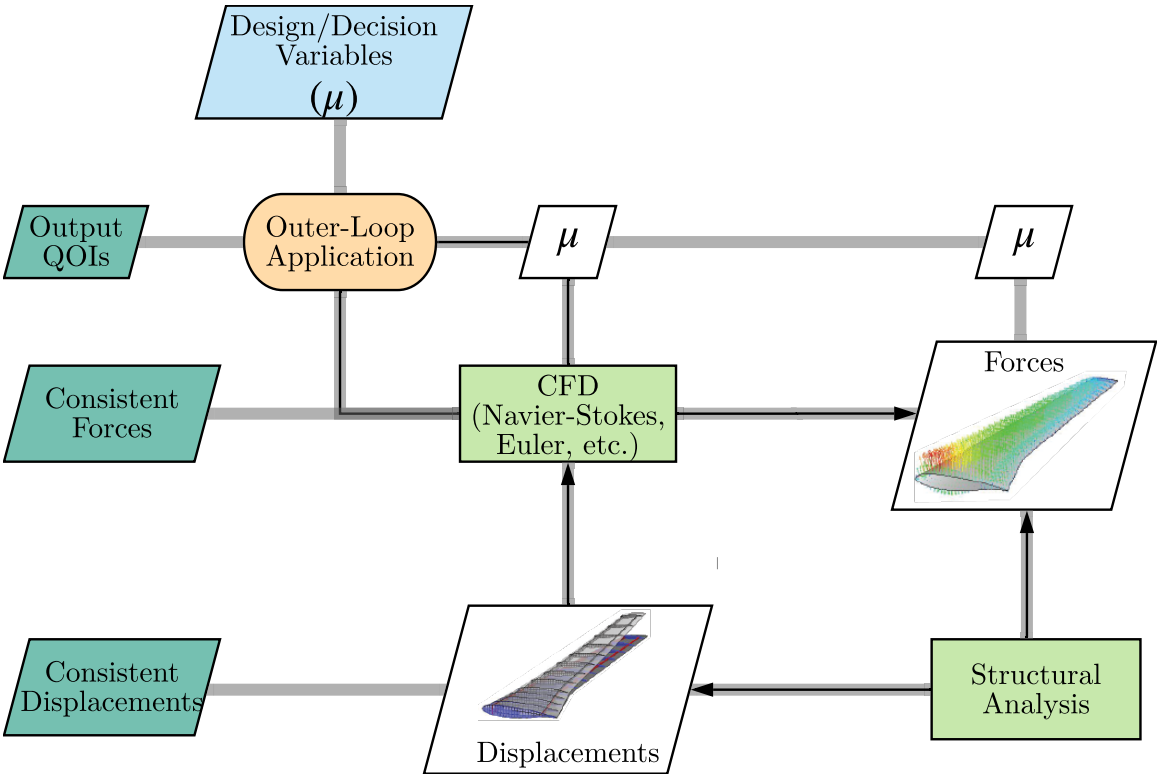


Figure 6: ROM Application Example: Coupling in Multidisciplinary Analysis

It is perfectly reasonable to assume that no definite answer exists to the question of which surrogate model works best for the problem in question. ROMs are known to be powerful only when the map from input to output can be sufficiently approximated by a low-rank subspace [16].

1.3.3 Machine Learning vs. Reduced Order Modeling - A Perspective

The broad and independently developed fields of Reduced order Modeling and Machine Learning have obvious commonalities. For instance, the determination of low-dimensional

subspaces (or manifolds) has close ties to techniques in machine learning, specifically dimensionality reduction [254]. However, the distinction arises due to a difference in vantage points throughout the respective fields' historical developments. ROMs have evolved with specific focus on reducing high-dimensional models of physical systems which are largely *physics-based*. On the other hand, machine learning has underpinnings in computational science. Specifically, focus is given to methods that create parsimonious models using data from some generating process (treated as a black-box). Although both fields share a common goal, ROM frameworks are inherently designed for physics-based formulations whereas machine learning disregards the actual process generating the data. In some cases, ROM formulations avail better predictive capabilities and provide certifiable error bounds. However, machine learning techniques are flexible and work under relaxed requirements (like unavailability of governing equations) which hold in practical situations. With time, the engineering community has started recognizing the power of fusing techniques from both fields. Consequently, we argue that the delineation between the fields has become somewhat fuzzy today.

1.4 Relevant Classifications and Successful Applications of Reduced Order Modeling

Conceding that ROMs are required to tackle the challenges posed by the paradigm-shift to simulation driven design to many-query settings, let us briefly discuss various classifications of ROMs that exist in literature. Based on some observations influenced by practical considerations, this section restricts the scope of this work building up to a statement presenting the main research objective in the following sections.

1.4.1 Parametric vs. Non-Parametric

Projection-based ROMs are valuable even in cases where parametric variation with respect to the design/decision variables is of no concern i.e. where only simulation and/or prediction of future behavior is/are of interest. The so-called non-parametric reduced order modeling methods have significantly matured over the years as evidenced by the textbooks by Antoulas [16] and Benner et. al. [29]. Their primary goal is sufficient reduction in the

size of a high-dimensional system. They are valuable when the simulation itself is computationally infeasible. *Real-time control* is another application area for non-parametric ROMs. In general, because controllers are as complex as the system they are designed to manipulate, ROMs provide solutions to the problems of storage (in devices), accuracy, and computational speed.

On the other hand, parametric reduced order modeling has only emerged recently (in the past two decades or so) as an important research area [31,158,222]. Parametric reduced order models play the role of surrogates where instead of scalars, fields or their evolution must be emulated rapidly as design/decision variables vary. Succinctly put, the goal of parametric ROMs is to generate inexpensive yet accurate models that emulate fields or scalar functions of fields over a range of parameter values. Note that in this document, the terms design/decision variables, parameters, and inputs are used interchangeably to refer to the parameters that the full order model depends on. Parametric dependence may appear in the form of boundary and/or initial conditions, geometric variables, coefficients in the governing equations etc. For further clarity, it is instructive to view parametric reduced order modeling as a means to create a family of non-parametric ROMs (at different samples of parameter points) equipped with some form of interpolation to enable fast prediction at unseen parameter values. Figure 7 shows a comparison between parametric and non-parametric ROMs.

Unique challenges must be overcome to handle parametric dependence with ROMs. The process of constructing ROMs requires evaluation of the expensive high-dimensional system. It stands to reason that building a ROM for every point in the parameter space in infeasible. The sensible approach involves parametrization of the set of ROMs built over a sampled set of points from the parameter space, which is the precise goal of parametric ROMs. In some cases, interpolation among ROMs relies on matrix interpolations [80]. Often times, properties of matrices being interpolated must be preserved at the interpolated parameter points [85]. Such additional steps add to the computational cost of evaluating the ROM online. Special care must be taken to ensure that the cost of online evaluation remains smaller than the original high dimensional system. In later chapters, we will look at specific

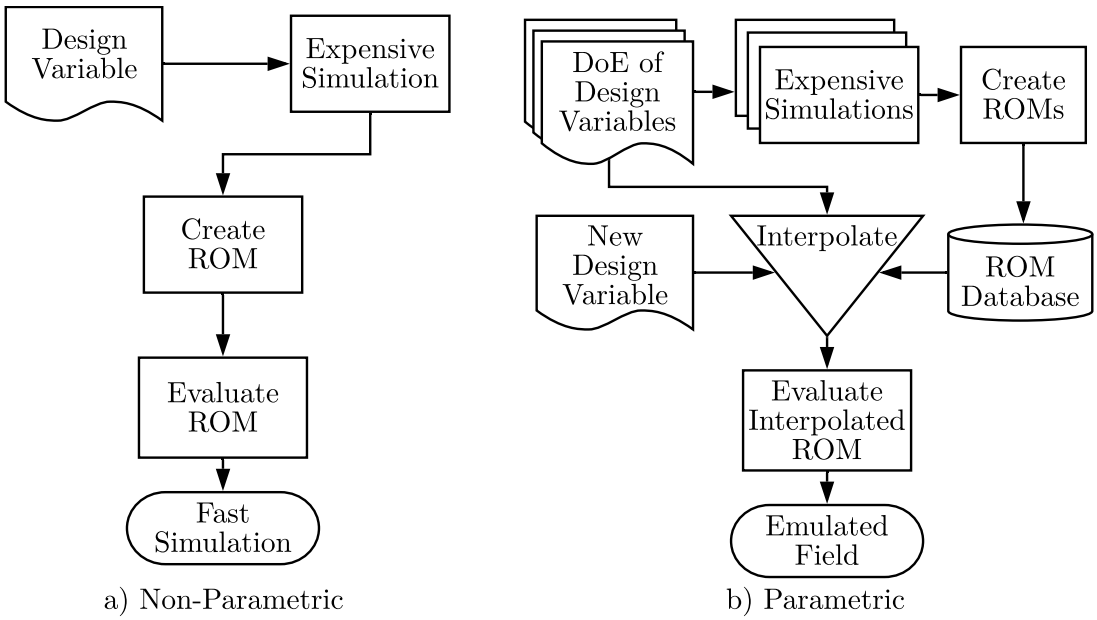


Figure 7: Difference Between Non-Parametric and Parametric ROMs

techniques to interpolate ROMs in detail.

1.4.2 Intrusive vs. Non-Intrusive

Another important distinction among ROMs - parametric and non-parametric - arises due to the inherent mechanism responsible for decreasing the degrees of freedom. The projection step ideally requires access to the full order operators. For the purpose of illustration, say, solving the high-dimensional full order model requires computing the solution of a linear system of equations describing the diffusion problem with a million unknowns. The projection step entails performing matrix multiplications (using projectors) from the left and right of the coefficient matrix (also called operator), resulting in a smaller linear system. The term *intrusive* signifies the requirement of availability/access to the original large matrix of coefficients to perform the reduction. Since developing software to solve systems arising from complex PDEs/ODEs is a coveted field of study in itself, the codes are often a technological commodity. Although open-source codes may be modified to provide access, such undertakings are often tedious. Unless the practitioner is developing source code from scratch i.e. in most practical cases, system operators are, or must be assumed to be inaccessible. Only data in the form of outputs (fields, their trajectories or scalar QOIs) and inputs

are available. ROMs developed to work under such constraints are called *Non-Intrusive*. Figure 8 depicts the difference between *intrusive* and *non-intrusive* ROMs at a high level.

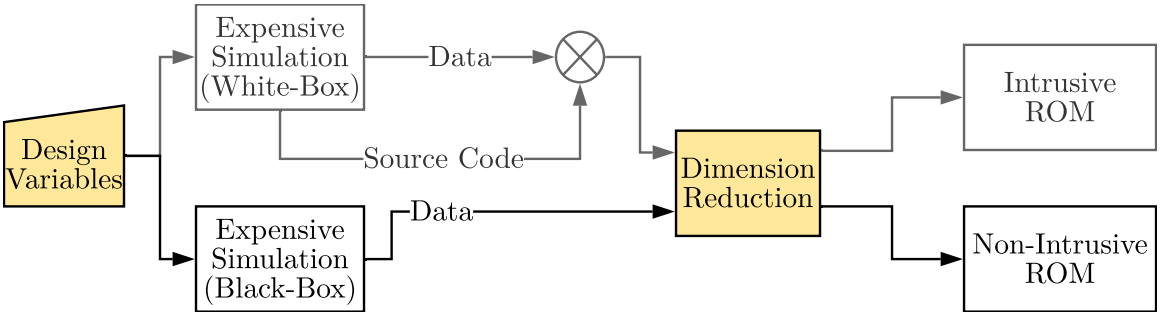


Figure 8: Difference Between Non-Intrusive and Intrusive ROMs

Let us now make note of some seminal articles that have successfully applied ROMs in the many-query context. Note that although the specific context may vary, all the applications relevant to the goal of this work deal with parametric ROMs. Some observations are made along the way leading to the introduction of the main objective of this dissertation.

1.4.3 Success in Construction of Parametric ROMs for Applications in the Many-Query Context

ROMs have been a subject of extensive evolution in the engineering domain. Perhaps one of the earliest applications in the aerospace community was two proposed decades ago by LeGresley et. al. [142]. Their work involved aerodynamic shape optimization using Proper Orthogonal Decomposition (POD) as a dimensionality reduction method in the reduced order modeling approach. Several other authors followed suit with applications in design space exploration [37,50,51,56,65,109,117,191,214,242], inverse design and data reconstruction [54], evaluation of aerodynamic loads [97], real-time structural assessment [163], rapid probabilistic analysis [52], large scale inverse problems [151], aeroelasticity [12, 152, 198], design optimization [7, 30, 33, 60, 63, 68, 81, 104, 146, 153, 184, 190, 197, 213, 224, 238, 248, 274, 275, 279], structures and solid mechanics [104, 278], earthquake engineering [24] etc. among many others.

Note that the applications mentioned above contain a mixture of both intrusive and non-intrusive methods to construct and employ parametric ROMs. The entire body of literature with applications and methods is too vast to be mentioned here. Interested readers may

refer to the exhaustive reviews by Ohlbeerger et. al. [183], Benner et. al. [31], and Lu et. al. [158]. These reviews contain overviews of novel methods and examples from the general engineering domain.

1.5 Challenges in Construction of Parametric ROMs

So far, the discussion has focused on the classifications of surrogate models and ROMs and their applications in many-query contexts. In this section, let us introduce some challenges that prohibit and severely limit the potential and promise ROMs offer for applications involving practical, large-scale engineering problems. The predictive performance of ROMs or any surrogate for that matter strongly depends on the size, quality and fidelity of the data used in the process of construction. For instance, it is a well-known fact that the quality of predictions deteriorates at points that are either farther away from or lie outside the envelope of observed data. In the case of ROMs, depending on the complexity of the physical phenomenon being mimicked, a large number of high-dimensional solutions may be required to capture the statistically rich physical features of the solution space, especially for large-scale practical problems. Therefore, the data requirements for accurate surrogate models for even moderately sophisticated problems can quickly become computationally burdensome. In the literature on surrogate models and ROMs, the *offline cost* of generating the data and training the model is typically compensated by the number of rapid *online* evaluations availed in the specific application. In fact, if one treats a surrogate model as a commodity that can be reused for several design problems, the justification of high *offline* costs becomes less of a consideration. However, because the main goal of ROMs and surrogate models is the replacement of expensive analyses by trading minimal accuracy for rapid evaluations, in this work it is argued that the challenges posed by practical, large-scale, and complex engineering problems in the context of constructing ROMs and surrogate models must be tackled to enable their easy adoption in engineering design.

This dissertation argues that there are two main unavoidable challenges associated with building surrogate models for practical engineering problems. One, the challenge posed by a large number of design/decision variables and two, the requirement to bear large

offline computational costs for both training the model and generating the training data. As alluded to above, complex problems by definition imply a big-data requirement which manifests as large upfront cost for generating the data using expensive analyses and data processing to train the model. Moreover, a telling trait of practical problems is a large design parametrization leading to several other challenges (as discussed below), in addition to the exacerbation of high *offline* cost.

1.5.1 High-Dimensional Decision/Design Variable Spaces

With the exception of a few intrusive ROM studies mentioned in the previous section, most deal with construction of parametric ROMs for problems that have only a couple of design/decision variables. However, a large number of design variables is unavoidable in most cases because: 1) for mature designs such as a traditional tube-and-wing, incremental improvements require a fine parametrization to explore the design space with appropriate resolution and 2) for advanced unconventional designs using analysis governed by PDEs, natural parametrizations may be high-dimensional. An example of the former is that of the aerodynamic shape optimization of wings (see figure 9) using the *free-form deformation* parametrization to generate shapes where the number of design variables can easily surpass $\mathcal{O}[100]$. For the latter, one can imagine a situation where the design variables are distributed in space (random diffusion coefficient function for the heat equation). In fact, for such cases, the number of design variables is technically equal to the number of state/field variables ($\mathcal{O}[10^{6-7}]$). The literature contains ample evidence regarding use of analyses with a high-dimensional design parametrization in the many-query context. Martins [166] presents results on three practical engineering design optimization problems (satellite design, aerodynamic and aero-structural problem) involving a large number of design variables. Kennedy et. al. [128] solve topology optimization problems with a number of distributed design variables in the $\mathcal{O}(10^{6-7})$ using gradient-based algorithms. While only a few examples are reported here, abundant examples are available in the literature. Note that most of these studies exploit parallelism to make computation time tractable (which may still be large). Moreover, the goal in these studies is not the creation of surrogate models for

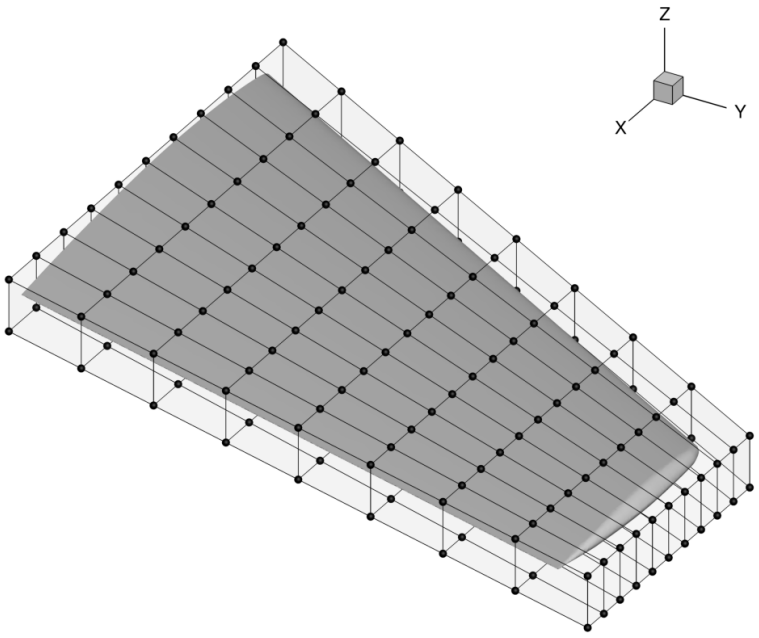


Figure 9: Example of a High-Dimensional Parametrization of the ONERA Wing (credit: Stanford Unstructured (SU2))

field outputs that allow rapid design space exploration. In this work, it is argued that the unique challenges associated with construction of surrogate models in high-dimensions is the primary reason for the lack of their adoption in many-query applications.

Many challenges exist with regard to creation of parametric ROMs (and surrogate models in general) in a high-dimensional parameter space [31,236,255]. Most of them are directly related to the *curse of dimensionality*, a broad, frequently used phrase to refer to problems associated with organization and analysis of data in high-dimensional spaces. In the specific context of dealing with a large number of input variables when constructing ROMs, some of the commonly known challenges are: 1) the sample size increases drastically (exponentially) with the number of design variables, 2) the volume of the design space increases rapidly with the number of design variables thereby requiring a disproportionately large number of additional samples for a sufficiently dense coverage of the space, 3) in addition to the associated computational cost, interpolation in higher dimensions for local quantities in ROMs can be challenging for the aforementioned reason, and 4) significance of variables

on the output (possibly high-dimensional, as for ROMs) is unknown a priori, i.e., the effect of some input dimensions may appear as noise in the output while other inputs may affect the response significantly. Therefore, the need for work on surrogates in high dimensional parameter spaces has been recently recognized by several researchers [154, 231, 232, 255], especially for parametric ROMs [31].

1.5.2 Conflict Between High Offline Cost and Accuracy

The two main contributors to the *offline* cost when training ROMs are the expensive evaluation time of the high-fidelity full order model and the processing costs for the resulting big dataset. Let us discuss the former followed by the challenges posed by the latter contributor.

The first challenge involves scenarios where a fixed computational budget for high-fidelity evaluations can severely limit the ability to generate enough data to achieve a model with sufficient predictive accuracy. This situation is more relevant and probable for expensive underlying analyses involving practical engineering problems. Therefore, this work argues that there is a need to address situations where high predictive accuracy is desired with a relatively small amount of high-fidelity data. A common family of techniques to deal with sparse data due to fixed computational budgets for generating high-fidelity data is termed as multi-fidelity modeling (as we discussed before). Multi-fidelity (MF) or variable-fidelity modeling leverages the correlation between analyses of varying fidelities and computational complexities in a synergistic manner to improve the predictive capability of the surrogate model [192] when only sparse high-fidelity data is available. The success of MF approaches hinges on a significant disparity between the computational costs of evaluating low- and high-fidelity full order models. When dealing with field outputs (the focus in this work), the literature on MF approaches consists of only a handful of studies [27, 37, 164, 195, 246]. While the fact that the discipline of parametric ROMs is incipient may seem to explain the sparsity of studies on multifidelity ROMs (MF-ROMs), it is in fact because analyses of different fidelities typically produce outputs with different dimensionalities, features, and topologies. For example, a simple technique for creating two data sources varying in fidelity is to run the same simulation on a coarse and a fine computational grid. By definition, the

size of the field outputs is inconsistent between the two simulations. Another example is the inconsistency arising due to the difference between the topology of geometries in varying levels of fidelity. In a structural analysis setting, for instance, the low-fidelity simulation may involve idealizing the structure as a shell or a simple beam. Even if the lengths of the outputs are identical, the inconsistency between the geometrical representations makes the field outputs from different fidelities incomparable. Reconciling and leveraging these inconsistencies is not trivial.

Even if ample data are available, the processing costs for large datasets on personal computers can be prohibitively high. Since the focus for this work is ROMs, the term data refers to a large matrix containing field outputs or another large matrix containing the inputs (or design variables/parameters) that generated the outputs. A large dataset is one that does not easily fit into memory of a modern personal computer and therefore limits the ability to perform quick computations of interest. Previous discussions argued that it is increasingly common for practical problems to have large field outputs. It will be shown later that an important step in the construction of ROMs consists in compressing these large datasets. It is therefore important to address the bottleneck caused by the inability to perform computations on modern personal computers especially to enable streaming, real-time and digital-twin applications with ROMs in-the-loop.

1.6 Summary

Growing compute power and progressions in numerical algorithm development have enabled simulation-driven design. Driven by the pressing need to produce new advanced unconventional air vehicles, there is a need to incorporate expensive high-fidelity tools early in the design process while retaining the conflicting ability to perform rapid many-query exercises. In other words, simulation-driven design processes incorporating computationally expensive tools must be available at affordable computational costs. Moreover, it was argued that modern day applications sometimes necessitate (or can benefit from) the prediction of high-dimensional fields as solutions rather than scalar outputs.

The inherent computational complexity for solving high-dimensional problems involving

fields can be circumvented by exploiting parametric ROMs. As discussed in section 1.3.2, ROMs avail several advantages that typical scalar surrogates lack. In some situations involving coupled analysis with outputs as fields in multidisciplinary settings, ROMs alone can enable computational tractability because they specifically emulate high-dimensional field outputs. In analyses involving time evolution, scalar surrogates cannot be modified organically while ROMs can. More importantly, as discussed above, the aforementioned situations cannot be avoided because unconventional concepts require high-fidelity analyses. The design setting inherently requires repeated evaluation of either a scalar function and/or a field output that depend(s) on design/decision variables. Therefore, methods that approximate fields (i.e. ROMs) are the only suitable option for enabling rapid design and analysis in the many-query context. Moreover, classifications in section 1.4 justify the focus on parametric non-intrusive ROMs in light of practical considerations. Non-intrusive ROMs compensate for the lack in accuracy relative to intrusive ROMs by providing the ability to work with black-boxes, essentially only requiring input-output pairs. Thus, the proposed work specifically focuses on non-intrusive parametric reduced order modeling methods and their applications in many-query engineering exercises. Before proceeding, let us consolidate and present the arguments to justify the focus on parametric non-intrusive ROMs for many-query contexts based on our observations so far.

Observations

1. ROMs are naturally required in situations involving prediction of high-dimensional fields where typical scalar surrogates fall short; for instance, when coupled disciplinary analyses are part of the many-query setting, ROMs alone can enable quick emulation of fields. Moreover, PDE/ODE system solves cannot be avoided when designing complex engineered systems with coupled analyses.

2. Even when emulation of scalar functions is required explicitly, ROMs are better suited because of their ability to retain the underlying mathematical structure of the system. Moreover, predicting the field instead of the scalar can allow computation of several scalars that depend on the field.

3. ROMs rely on data-compression techniques to find a compact, low-dimensional subspace thereby automatically considering the spatial and temporal relationships within fields.

4. ROMs can organically handle time evolution of high-dimensional fields whereas data-fit scalar surrogates find it challenging, often leading to a loss in flexibility.

5. Albeit intrusive approaches are relatively well-suited for reduced order modeling in comparison to non-intrusive methods, their applicability is hindered due to practical reasons; such as unavailability of the source code, requirement of tedious rework etc.

6. High-dimensional design variable spaces (often characterizing practical problems) in the context of ROMs (and surrogates, in general) have received limited attention.

7. The two main contributors to high *offline* costs associated with the construction of ROMs for complex problems viz. limited high-fidelity data under a fixed computational budget and high processing costs for large datasets (matrices) are major challenges that limit the potential of ROMs for practical problems.

The observations listed above serve as high-level reasons to justify focus on non-intrusive parametric ROMs. Deferring details on the specific deficiencies in the state-of-the-art to later discussions in chapter 2, let us now introduce the motivating question that inspires the work in this dissertation.

Motivating Question

How can *non-intrusive parametric reduced order modeling methods* be improved to enable the creation of surrogates to approximate *high-dimensional fields* emanating from *PDE-based black-boxes* to enable *many-query* exercises encountered in *practical* engineering design and analysis problems?

For readers well-versed with ROMs, the overall research objective premised on the high-level observations/arguments made above and detailed observations made in chapter 2 may be precisely stated as follows:

Research Objective

Develop and/or enhance *non-intrusive parametric reduced order modeling methods* to address challenges that limit their viability in many-query contexts involving practical engineering design problems with a focus on analyses that:

1. are expensive to evaluate,
2. output high-dimensional field quantities, and
3. are characterized by a relatively large number of design variables.

1.7 Document Outline

The following chapters will clarify reasons for pursual of the precise research objective stated above by drawing from detailed observations in the literature. The remainder of the document is organized as follows:

- *Chapter 2* sets up the mathematical background underpinning the ROM literature

along with the ingredients required to construct parametric ROMs. Intrusive ROMs are briefly introduced followed by an in-depth presentation of non-intrusive methods (the focus in this work). Then, the three challenges namely large analysis execution time under a fixed budget, large datasets, and high-dimensional input spaces along with efforts that have attempted to solve them are presented in the context of ROMs. Observations are made throughout the exposition to justify the main research question stated above and scope the subsequent work.

- *Chapter 3* presents the core research areas and builds the accompanying questions that will answer the motivating question and fulfill the objective of this research. Note that in this chapter, justification and presentation of relevant questions will be the main focus. While it may be chronologically logical to introduce some questions later, they are first presented as a wholesome unit before delving into the details in subsequent chapters.

- *Chapter 4* presents the development of methods that address the challenge related to construction of ROMs using large datasets. The chapter presents details of the experiments performed on a canonical problem to answer specific aspects of the techniques that the proposed method relies on for success. Finally, all the experiments are repeated on practical application problems to demonstrate and highlight the benefits and point out any deficiencies.

- *Chapter 5* presents the development of a method to address the challenge related to large input spaces by discovering a low-dimensional input subspace. This chapter also investigates various aspects for successful resolution of the challenge posed by large input spaces by performing some experiments to answer relevant research questions posed in chapter 3 first on a canonical problem followed by two large-scale application problems. Throughout the presentation, the demonstrations highlight both the benefits and inadequacies of the method by comparing it to a relevant contemporary baseline method.

- *Chapter 6* presents the final investigation pursued in this dissertation. It develops

and tests a multifidelity method to discover a low-dimensional input subspace using data from both a high- and a low-fidelity dataset. This method is first applied on well-known engineering functions followed by an application on one practical subsonic airfoil flow problem.

- *Chapter 7* culminates this work by briefly consolidating and summarizing the findings and contributions of this dissertation. Finally, some avenues for future work and scope for improvement based on the lessons learnt are presented.

CHAPTER II

BACKGROUND AND LITERATURE

In this chapter, let us first briefly discuss the mathematical framework for reduced order modeling. We will then give a broad introduction of dimensionality reduction because it is arguably the most fundamental ingredient in the efficient treatment of high-dimensional data in machine learning and reduced order modeling. An in-depth review of non-intrusive parametric ROMs will then follow. The discussion will conclude with identification of areas that have received limited attention.

2.1 Projection-based Parametric Reduced Order Modeling

In this section, we give a succinct presentation of the mathematical setting and formulation of reduced order modeling methods. The explanation aims at precisely clarifying the difference between intrusive and non-intrusive ROMs. Since the reasons for the pursuit of non-intrusive methods have been established, an in-depth review of state-of-the-art techniques will follow a brief introduction of the general framework.

Parametric Reduced Order Models are touted as physics-based approximation techniques that provide a computationally inexpensive low-dimensional emulator for an expensive model of physical laws. The original high-dimensional system is also called the full order model (FOM). They are physics-based in the sense that spatial and temporal relationships between the discretized field quantities are retained in their formulation. They achieve reduction in computational cost by decreasing the number of unknowns that need to be solved for as compared to a large FOM. All the reduction methods operate under the assumption that most of the variation in the solution space can be captured by low-dimensional subspace.

2.1.1 The General Framework

The setting in which ROM methods are developed begins with the spatial and/or temporal discretization of partial differential equations. Precisely speaking, this process leads to large finite-dimensional system of ordinary differential equations (ODEs). Any general spatially discretized (using for instance, finite element or finite difference schemes) time continuous, parametric system can be expressed as

$$\begin{aligned} \frac{d\boldsymbol{w}(t)}{dt} &= \boldsymbol{f}(\boldsymbol{w}(t), t, \boldsymbol{\mu}) \\ \boldsymbol{w}(t_0, \boldsymbol{\mu}) &= \boldsymbol{w}_0 \end{aligned} \tag{1}$$

where $t \in \mathbb{R}_+$ and $t \geq t_0$ denotes time, $\boldsymbol{\mu} \in \mathbb{R}^d$ represents the design/decision variables that may appear as the boundary conditions, shape parameters etc., $\boldsymbol{w} \in \mathbb{R}^m$ denotes the discretized state variable or field quantity (eg. pressure, deflections, stress etc.) that must be solved for, and $\boldsymbol{f} : \mathbb{R}^m \times \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^m$ is the nonlinear function that is a resultant of the discretization process. As mentioned before, m is the size of the discretized field variable in the spatial domain. Depending on the demand dictated by complexity of physics being captured or the number of spatially distributed quantities being computed, m can be as large $\mathcal{O}[10^{6-8}]$. As one may imagine, rightly so, solving for unknowns in that order can be daunting. Some authors often present equation (1) in the residual form which is consistent with the presentation of root finding problems. If the system is steady for instance, the residual form is

$$\boldsymbol{r}(\boldsymbol{w}, \boldsymbol{\mu}) = \boldsymbol{0} \Leftrightarrow \boldsymbol{f}(\boldsymbol{w}(t), t, \boldsymbol{\mu}) = \boldsymbol{0},\ \boldsymbol{r} : \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R}^m \tag{2}$$

In the unsteady time-continuous case, the residual form can be expressed as

$$\boldsymbol{r}(\boldsymbol{w}(t), \boldsymbol{\mu}, t) = \boldsymbol{0} \Leftrightarrow \frac{d\boldsymbol{w}(t)}{dt} - \boldsymbol{f}(\boldsymbol{w}(t), t, \boldsymbol{\mu}) = \boldsymbol{0},\ \boldsymbol{r} : \mathbb{R}^m \times \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^m \tag{3}$$

whereas if the system is unsteady and space-time discrete, at every time-step (say, n) a residual set of equations must be solved. This is usually expressed as

$$\begin{aligned} \boldsymbol{r}^n(\boldsymbol{w}(t_n), \boldsymbol{w}(t_{n-1}), \ldots, \boldsymbol{w}(t_{n-l}), \boldsymbol{\mu}) = \boldsymbol{0},\ \text{where } \boldsymbol{r}^n : \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R}^m \\ n = 1, 2, \ldots, N \text{ where } N \in \mathbb{N} \end{aligned} \tag{4}$$

where N is the number of time steps due to the temporal discretization and l is dictated by the order of accuracy of the time integration scheme. Equations (2), (3), and (4) encapsulate the various forms that can be taken by the parametric-PDEs converted into ODEs. The fundamental assumption that reduced order modeling formulations make is the solution $\boldsymbol{w}$ lies in a subspace of dimension $k \ll m$. This assumption is enforced by restricting the solutions to a $k-$dimensional subspace of $\mathbb{R}^m$ through the following *ansatz*:

$$\boldsymbol{w} = \boldsymbol{\Phi}(\boldsymbol{\mu})\boldsymbol{w_r}(t) \tag{5}$$

where $\boldsymbol{w}_r \in \mathbb{R}^k$ denotes the coordinates in the *reduced space* (hence the subscript r) $\boldsymbol{\Phi} \in \mathbb{R}^{m\times k}$ called the *trial subspace*. This terminology is commonly found in texts on *Reduced Basis Methods* (RBMs) [183]. ROMs fall under the general category of RBMs, the difference being that RBMs have a foundation in the theory of solving differential equations at the continuous (variational) level using the concept of *weak formulation* [111].

Reduction in the number of unknowns is achieved by projecting the set of equations onto the space spanned by another subspace (say, $\boldsymbol{\Psi} \in \mathbb{R}^{m\times k}$) called the *test subspace*. Precisely speaking, the reduction is achieved by substituting equation (5) for $\boldsymbol{w}$ into the residual form and enforcing its orthogonality to the *test subspace*. Note that both $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ may be functions of $\boldsymbol{\mu}$ but for simplicity in exposition, we assume that they are valid subspaces over the entire parameter domain. The substitution similarly reduces the unknowns in the time-continuous formulation (eq. (3)) to

$$\begin{aligned} &\boldsymbol{\Psi}^T\boldsymbol{r}(\boldsymbol{\Phi}\boldsymbol{w}_r(t), \boldsymbol{\mu}, t) = \boldsymbol{0} \\ &\Leftrightarrow \boldsymbol{r}_r(\boldsymbol{w}_r(t), \boldsymbol{\mu}, t) = \boldsymbol{0};\ \boldsymbol{r}_r = \boldsymbol{\Psi}^T\boldsymbol{r} \end{aligned} \tag{6}$$

Notice that instead of solving for $\boldsymbol{w}$, we must now solve: 1) a smaller set of equations ($\boldsymbol{r}_r$) and 2) for a smaller number of variables ($\boldsymbol{w}_r$). The relationship between *test* and *trial* subspaces leads to two different methods.

Galerkin and Petrov-Galerkin Projections

Under the assumption that both the *trial* and *test* are identical to each other (i.e. $\boldsymbol{\Phi} = \boldsymbol{\Psi}$),

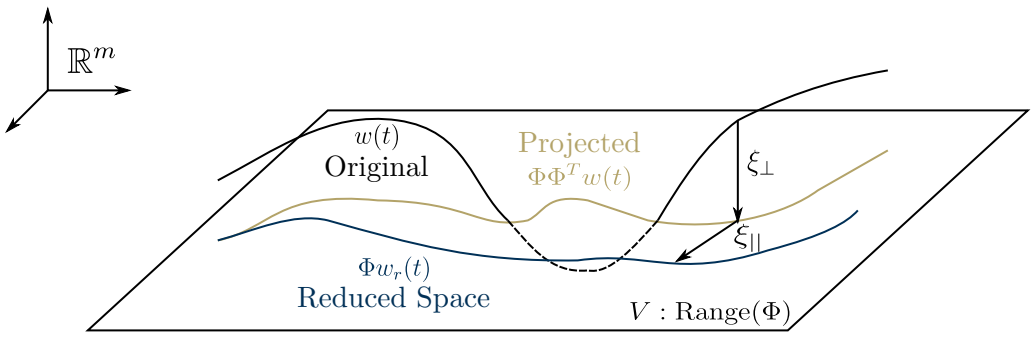


Figure 10: Error due to Galerkin Projection (adapted from [210])

we have the *Galerkin* method [53]. The reduced set of equations can be written as

$$\boldsymbol{\Phi}^T \boldsymbol{r}(\boldsymbol{\Phi} \boldsymbol{w}_r(t), \boldsymbol{\mu}, t) = \mathbf{0} \tag{7}$$
$$\boldsymbol{r}_r = \boldsymbol{\Phi}^T \boldsymbol{r}$$

On the other hand, when $\boldsymbol{\Phi} \neq \boldsymbol{\Psi}$ or the *trial* and *test* subspaces are different, a common approach is to find $\boldsymbol{\Psi}$ such that the reduced order residual is minimized. Such a projection strategy is called the *Petrov-Galerkin* method [59].

$$\boldsymbol{\Psi}^T \boldsymbol{r}(\boldsymbol{\Phi} \boldsymbol{w}_r(t), \boldsymbol{\mu}, t) = \mathbf{0} \tag{8}$$
$$\Leftrightarrow \mathbf{0};\ \boldsymbol{r}_r = \boldsymbol{\Psi}^T \boldsymbol{r}$$

The reduction procedure gives rise to a small system of equations that may be solved either as a nonlinear least-squares or a nonlinear root finding problem [43]. Notice that we have not explicitly seen how the subspaces are chosen yet. The only requirement is that the projection subspaces must be valid projectors mathematically (i.e. must be full rank and preferably orthogonal). For a comparison between Galerkin and Petrov-Galerkin projections, interested readers are referred to the work by Carlberg et. al. [58]. Figure 10 shows the error introduced due to Galerkin projection and enforcement of the ansatz. Note that there are two components of the error i.e. the parallel and perpendicular contributions respectively. Curious readers who may ponder about the need for space $\boldsymbol{\Psi}$ to project can imagine how projecting in a subspace that is different from $\boldsymbol{\Phi}$ can decrease error in the reduced coordinates. An approach that capitalizes on this possibility especially in the context of ROMs was put forth by Carlberg et. al. [59]. They show that in order for the

reduced order system to be consistent with the original full order model while minimizing the reduced residual, the optimal projection spaces (matrices) depend on the properties of the original system [62]. In a series of papers [57, 59, 61], they show that while minimizing the reduced residual and maintaining consistency, the *trial* and *test* subspaces are identical i.e. $\boldsymbol{\Phi} = \boldsymbol{\Psi}$ if the jacobian of the full system is symmetric positive definite, and $\boldsymbol{\Psi} = \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\mu}}\boldsymbol{\Phi}$ otherwise. The actual savings in computational costs and the precise reason for the existence of non-intrusive and intrusive ROMs will now be elucidated through an example for a static linear system.

Consider a parametric linear system that arises from the spatial discretization of a linear PDE (Poisson's equation, for instance),

$$\boldsymbol{A}(\boldsymbol{\mu})\boldsymbol{w} = \boldsymbol{b};\ \boldsymbol{A} \in \mathbb{R}^{m\times m} \text{ and } \boldsymbol{w}, \boldsymbol{b} \in \mathbb{R}^{m\times 1}$$

Assume that $\boldsymbol{b}$ contains all the boundary conditions and any generic RHS in the PDE. The minimum residual *Petrov-Galerkin* projection for this system [51] yields $\boldsymbol{\Psi} = \boldsymbol{A}\boldsymbol{\Phi}$. Once $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ are computed appropriately, the ansatz $\boldsymbol{w} = \boldsymbol{\Phi}\boldsymbol{w}_r$ is substituted in the original system. Following the projection, we get

$$\begin{aligned}
\boldsymbol{\Phi}^T\boldsymbol{A}(\boldsymbol{\mu})^T\boldsymbol{A}(\boldsymbol{\mu})\boldsymbol{\Phi}\boldsymbol{w}_r &= \boldsymbol{\Phi}^T\boldsymbol{A}^T\boldsymbol{b}(\boldsymbol{\mu}) \\
\text{or, } \boldsymbol{A}_r(\boldsymbol{\mu})\boldsymbol{w}_r &= \boldsymbol{b}_r(\boldsymbol{\mu}) \\
\text{where, } \boldsymbol{A}_r &= \boldsymbol{\Phi}^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{\Phi} \\
\text{and, } \boldsymbol{b}_r &= \boldsymbol{\Phi}^T\boldsymbol{A}^T\boldsymbol{b}
\end{aligned}$$

It can be clearly seen that instead of solving for a system with m unknowns, a smaller system with $k \ll m$ unknowns must be solved for. In precise terms, this is the computational advantage offered by ROMs. Figure 11 shows a pictorial depiction of the reduction in number of unknowns due to projection for a linear system. Note that the dependence of $\boldsymbol{A}$, $\boldsymbol{A}_r$, $\boldsymbol{b}$, and $\boldsymbol{b}_r$ on $\boldsymbol{\mu}$ has been dropped in the figure for convenience. The projection step and consequently, solution of the reduced equations requires access to the system matrix $\boldsymbol{A}$ at every parameter value $\boldsymbol{\mu}$. This is the reason that these methods are called *intrusive* in nature because access to system operators requires the ability to either explicitly access $\boldsymbol{A}$ or

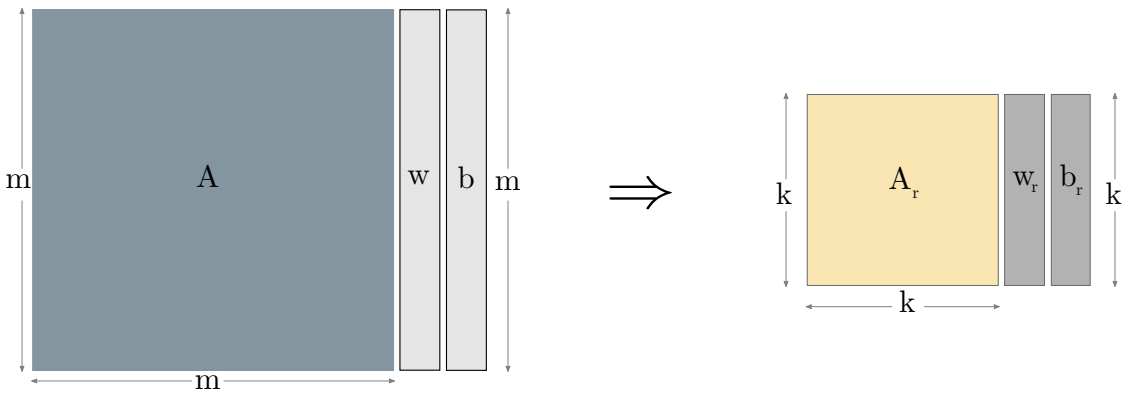


Figure 11: Example of Reduction for Linear Systems

perform matrix-matrix and matrix-vector products given the projection subspace matrices ($\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$). In contrast, for the *non-intrusive* case, we only have access to either the system states or the outputs (that are functions of state and parameters).

A general nonlinear RHS in equation (6) poses additional challenges to reduction in computational cost. Observe that the arguments accepted by the reduced residual equation contain $\boldsymbol{\Phi}\boldsymbol{w}_r$. Say Newton's method is used to find the roots of this equation. At every iteration the formation of the reduced residual still scales as $\mathcal{O}[m]$, as such limiting the computational savings due to projection. The literature consists of a variety of techniques to overcome this problem. Gappy POD [91,262], Missing Point Estimation (MPE) [19,283], Discrete Empirical Interpolation Method (DEIM) [64, 180] and Best Points Interpolation Method (BPIM) [116,181] are some of the well-known techniques. These methods recognize a low-rank structure in the high-dimensional function and exploit it by defining an appropriate basis and a set of 'most important/optimal/best' interpolation points. The selection of interpolation points is posed as a least-squares minimization problem for full function reconstruction confined to the low-dimensional basis. Each method essentially differs in the conditions/criteria used for defining the best set of interpolation points. Recently, the Dynamic Mode Decomposition (DMD) has been used to perform reduction in the cost of evaluating the nonlinear term [9]. However, the approach is not parametric and is only applicable to time dependent systems at a specific parameter point. Another key observation is in recognizing that these methods are fruitful only if the nonlinearity can be exclusively

evaluated at certain points; as determined by the algorithm selecting the 'best' interpolation points. Therefore, unless practitioners have control over the solver to specify the points at which to evaluate the nonlinear term, these methods may not perform as expected in terms of the computational savings they promise to offer. They are still useful in selecting "sensor" locations such that the error in reconstruction is minimum. In this sense, the algorithms resemble developments in the vast literature of compressive sensing [208].

For the sake of completeness, the general process of using ROMs instead of scalar surrogates is shown in figure 12 to aid the discussions that follow. The discussion so far lacks two important ingredients, namely, computation of the *test* and *trial* subspaces and handling parametric dependence (i.e. design/decision variable dependence). These two topics are subjects of the following sections.

2.2 Dimension Reduction: A Brief Introduction

High-dimensional data is ubiquitous in a variety of problems. The digital representation of an image is perhaps the simplest example. Depending on the resolution, accurate representation of an image can easily require in the order of a million ($\mathcal{O}[10^6]$) numbers denoting the intensity of color. In image classification exercises, operating with a dataset of such digital representations can become problematic. Speech signals, genome sequences, digitized documents, and internet searches are other examples of high-dimensional data. When dealing with a large number of degrees of freedom (DOFs), statistical and machine learning techniques face the so-called *curse of dimensionality*. Proper handling and understanding of large data therefore require reduction in its dimensionality. Besides, it is often the case that most naturally generated data intrinsically lie in lower-dimensional spaces. The primary role of dimensionality reduction methods is the transformation of high-dimensional data into a meaningful and reasonable representation of reduced dimensionality [254]. According to Sorzano et. al. [240], all methods designed to reduce dimensionality hinge on either of the following assumptions: 1) only a subset of the DOFs are significant, or 2) the dataset is redundant in the sense that some dimensions are interdependent due to the coherence amongst members (data points) in the set. The terms *Dimensionality Reduction*, *Data*

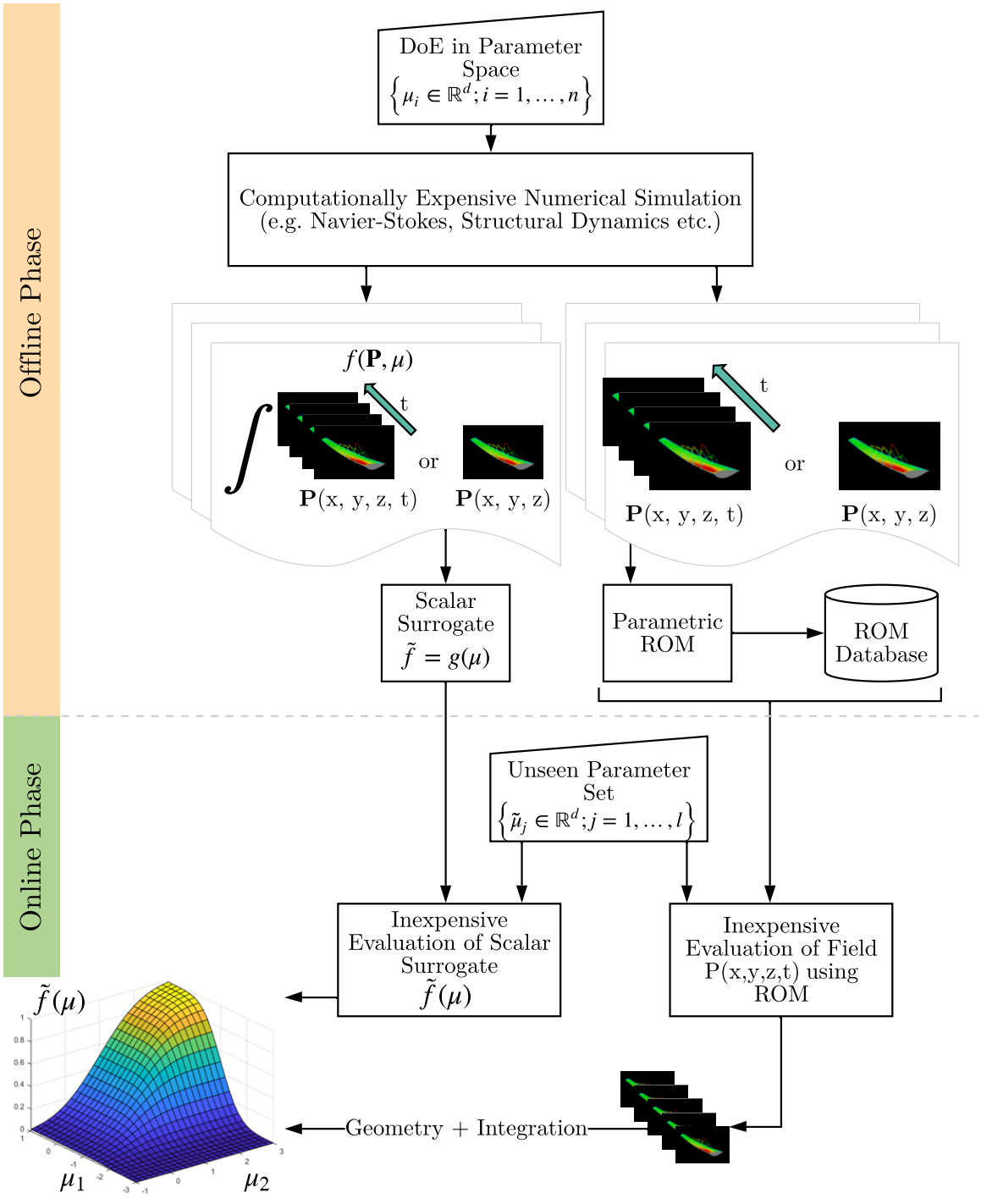


Figure 12: Comparison of ROMs and Scalar Surrogates

Compression, and *Feature Selection* are used interchangeably when referring to the general process.

In concrete terms, given dataset $\boldsymbol{X} : \{\boldsymbol{x}_i \in \mathbb{R}^m\}_{i=1}^{i=n}$ with n samples in m-dimensional space, the objective is to find a mapping $\boldsymbol{\phi} : \boldsymbol{x} \in \mathbb{R}^m \mapsto \boldsymbol{y} \in \mathbb{R}^k$ through which a faithful lower-dimensional representation $\boldsymbol{Y} : \{\boldsymbol{y}_i \in \mathbb{R}^k\}_{i=1}^{i=n}$ of $\boldsymbol{X}$ can be obtained. If such a compression is accurate, k is called the effective or intrinsic dimensionality of the data. Ideally, it is desired to have $k \ll m$. The space into which $\boldsymbol{\phi}$ maps high-dimensional data points is also referred to as the *latent space*. The taxonomy of techniques for performing data compression classifies them into *linear* vs. *nonlinear* methods and *supervised* vs. *unsupervised* methods.

2.2.1 Supervised versus Unsupervised

The latter distinction differentiates techniques based on the presence or absence of descriptive label information along with the high-dimensional data. While supervised methods use the relationship between label information and data to construct $\boldsymbol{\phi}$, unsupervised methods exclusively exploit the relationship amongst data points based on some notion of distance. Effectiveness of both supervised and unsupervised methods depends on a rich representative sampling of the full space. We will see later that unsupervised dimension reduction techniques play a major role in computing the subspaces for performing the reduction in the number of discretized unknowns in the context of ROMs.

2.2.2 Linear versus Nonlinear

As their name suggests, linear methods reduce dimensionality by constructing a linear transformation between spaces. In other words, $\boldsymbol{\phi}$ is a linear map. Nonlinear methods are known to outperform linear ones in compressing data sampled from curved spaces (also called manifolds) into lower dimensional manifolds.

A mathematical artifact of nonlinear methods is that the map $\boldsymbol{\phi}$ and inverse map $\boldsymbol{\phi}^{-1}$ are implicit. In practice, this trait is disadvantageous because the mapping operation must either be approximated or completely repeated when adding new points. In addition, the full space dataset cannot be recovered for a new point in the latent space without repetition

of the entire process. Work by Franz et. al. [98, 107] exemplifies this issue in the context of ROMs. They use a nonlinear method for creating a non-intrusive parametric ROM where the inverse map ϕ^{-1} is approximated using interpolation. On the contrary, with linear methods the map and inverse map are matrix multiplication operations which are relatively straightforward. Other researchers [132] have shown that when cardinality of the dataset is smaller than the number of dimensions, nonlinear methods find a lower dimensional linear subspace. Moreover, nonlinear methods are known to outperform linear methods only on contrived (i.e. artificially designed to elucidate nonlinearity) data sets [254]. Finally, it is not uncommon to use the so-called *kernel trick* [39] in conjunction with linear techniques to handle data belonging to nonlinear manifolds.

The discussion above shows that although nonlinear dimensionality reduction methods serve well in certain situations, linear methods are sufficient for use with ROMs. This can be evidenced by the successful applications from literature shown in chapter 1. In fact, recent work [79] has shown that while nonlinear dimension reduction methods are relatively more accurate in capturing localized nonlinearities, linear methods result in a smaller overall reconstruction error when used for constructing nonintrusive parametric ROMs. We assert that the simplicity and flexibility of linear methods make them a popular choice among practitioners of ROMs. Specifically, Proper Orthogonal Decomposition (POD) (identical to Principal Component Analysis (PCA)) has undoubtedly emerged as a favorite choice for dimension reduction in the ROM community, especially for non-intrusive methods.

2.2.3 Basis Computation Methods

For the sake of comprehensiveness, we briefly discuss basis computation methods applicable to both intrusive and non-intrusive methods. According to Benner et. al. [31], strategies to compute bases can be categorized into 1) Tangential Interpolation, 2) Proper Orthogonal Decomposition, and 3) Balanced Truncation. Applicability of balanced truncation and tangential interpolation is relevant in the context of parametric linear time invariant control systems where emulation of the output (rather than the state) is important. Balanced Truncation exploits system theoretic concepts of reachability and observability Gramians. The

Gramians are obtained by solving the Lyapunov equations [32]. The main idea is to neglect small singular values of the transfer function (input-output map). Roughly speaking, basis sets computed via balanced truncation eliminate states corresponding to negligible singular values of the transfer function. Note that balanced truncation basis sets are computed at every parameter point and interpolated at new parameter points. Naturally posed in the frequency domain, rational interpolation computes basis sets by interpolating the transfer function (dependent on $\boldsymbol{\mu}$ and frequency ω) of the reduced and full order system at a set of parameter ($\boldsymbol{\mu}_i$, $i = 1, \dots$) and frequency (ω_j, $j = 1, \dots$) values along a certain direction. Moreover, the derivatives w.r.t. ω and $\boldsymbol{\mu}_i$ can also be matched. The rationale behind this approach is the optimal reduction in some norm of transfer function. For an in-depth discussion, refer to the survey by Benner et. al. [31] and the references therein.

While the methods discussed above are used widely in the controls community, they either work under restrictive assumptions about the system type or are intrusive in nature. The methods are either exclusively applicable to dynamical systems or require that the underlying system be linear time invariant. On the other hand, the proper orthogonal decomposition does not depend on any such assumptions. This is one of the main reasons for its fame and ubiquitous use with ROMs. Let us now discuss its development in-depth as it pertains to non-intrusive parametric reduced order modeling.

2.2.4 Proper Orthogonal Decomposition

This popular method is known by a wide variety of names such as Principal Component Analysis (PCA) or Hotelling Analysis [119], Empirical Component Analysis [155], Proper Orthogonal Decomposition (POD) [160], Quasiharmonic Modes [44], Singular Value Decomposition [105], Empirical Eigenfunction Decomposition [235] in different fields. Its development is originally attributed to independent work by Karhunen [125] and Loéve [174] giving it the name *Karhunen-Loéve decomposition*. It was introduced in fluid dynamics by Lumley [160] as a technique for extraction and characterization of coherent turbulent structures. In a nutshell, the method is a numerical algorithm to objectively decompose data into a minimal set of modes or basis functions to capture as much energy as possible [243].

In other words, given some data, the algorithm finds a set of directions that captures the majority of the variance in the data. For ROMs - in the same spirit - it yields a subspace that optimally describes a given set of data [219].

Say. we start with the state ($\boldsymbol{w}(\boldsymbol{x},t) \in \mathbb{R}^m$) defined over a discretized spatial domain $\boldsymbol{x} \in \mathbb{R}^{1,2,\text{ or }3}$ and discrete time instances t_i. The formulation for POD seeks an orthogonal decomposition of the form

$$\boldsymbol{w}(\boldsymbol{x},t) - \bar{\boldsymbol{w}}(\boldsymbol{x}) = \sum_j a_j(t)\boldsymbol{\phi}_j(\boldsymbol{x}) \tag{9}$$

where $\boldsymbol{\phi}_j \in \mathbb{R}^m$ are a set of orthogonal spatial modes and $a_j \in \mathbb{R}$ are scalar expansion coefficients. Note that it is assumed that a separation of variables exists and is possible. This may not be true in most situations and depends on the properties of the system and type of information we wish to extract [118]. The so-called classical POD starts with the following data set containing snapshots of the field variables stacked as a matrix (the argument $\boldsymbol{x}$ has been dropped for convenience).

$$\tilde{\boldsymbol{w}}(t) = \boldsymbol{w}(\boldsymbol{x},t) - \bar{\boldsymbol{w}}(\boldsymbol{x}) \in \mathbb{R}^m, \quad t = t_1, \ldots, t_n \tag{10}$$

$$\boldsymbol{W} = [\tilde{\boldsymbol{w}}(t_1), \tilde{\boldsymbol{w}}(t_2), \ldots, \tilde{\boldsymbol{w}}(t_n)] \in \mathbb{R}^{m\times n} \tag{11}$$

$\boldsymbol{W}$ is the data/snapshot matrix. The goal of POD is to compute a set of orthogonal vectors (directions in $\mathbb{R}^m$) which can optimally represent any snapshot $\boldsymbol{w}(\boldsymbol{x},t)$ using the least number (low-rank) of directions/modes. An obvious, seldom stated assumption is that high variance directions contain more information than other directions. In mathematical terms, the objective can be stated as: find principal directions such that projection of the data matrix onto these directions maximizes the captured variance in the data. This concept is equivalent to the optimization of *Rayleigh Quotients*. It is also related to the low-rank approximation of a matrix [83]. In precise terms, the solution can be obtained by solving the following eigenvalue problem

$$\begin{aligned} &\boldsymbol{R}\boldsymbol{\phi}_j = \lambda_j\boldsymbol{\phi}_j,\ \boldsymbol{\phi}_j \in \mathbb{R}^m,\ \lambda_j \geq \lambda_{j+1} \geq 0,\ j = 1, \ldots, n-1 \\ &\boldsymbol{R}\boldsymbol{\Phi} = \boldsymbol{\Phi}\boldsymbol{\Lambda},\ \boldsymbol{\Phi} \in \mathbb{R}^{m\times m},\ \boldsymbol{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_m) \end{aligned} \tag{12}$$

Here $\boldsymbol{R}$ is the *covariance* matrix of the vector ensemble $\boldsymbol{W}$ defined as

$$\boldsymbol{R} = \sum_{i=1}^{n} \tilde{\boldsymbol{w}}(t_i)\tilde{\boldsymbol{w}}(t_i)^T \tag{13}$$

$$\text{or, } \boldsymbol{R} = \boldsymbol{W}\boldsymbol{W}^T \in \mathbb{R}^{m\times m} \tag{14}$$

In fact, the actual definition of covariance involves a division of $\boldsymbol{R}$ by (n) or $(n-1)$. When stated as in equation (13), it is assumed here that the factor is accounted for in the eigenvalues λ_i. The eigenvectors or the so-called POD modes are orthogonal and satisfy

$$\langle \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \rangle = \int_{\boldsymbol{x}} \boldsymbol{\phi}_i \,.\, \boldsymbol{\phi}_j d\boldsymbol{x} = \delta_{ij}, \quad j,k = 1,\ldots,m \tag{15}$$

The eigenvalues are used to compute the *Relative Information Content* (RIC) to determine the truncation rank. Large eigenvalues correspond to directions that explain relatively higher variance. Using the following condition, only k-modes are chosen to reconstruct the field.

$$\text{RIC} = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{j=1}^{m} \lambda_j} \approxeq 1 \tag{16}$$

Once k is fixed appropriately, the state or field can be represented using the truncated expansion given by eq. (17). The time dependent reduced coefficients (coordinates in the latent space) can be computed using eq. (18).

$$\boldsymbol{w}(\boldsymbol{x},t) - \bar{\boldsymbol{w}}(\boldsymbol{x}) = \sum_{j=1}^{r} a_j(t)\boldsymbol{\phi}_j(\boldsymbol{x}) \tag{17}$$

$$a_j(t) = \langle \boldsymbol{w}(\boldsymbol{x},t) - \bar{\boldsymbol{w}}(\boldsymbol{x}), \boldsymbol{\phi}_j(\boldsymbol{x}) \rangle = \langle \tilde{\boldsymbol{w}}(t), \boldsymbol{\phi}_j \rangle \tag{18}$$

Note that the magnitude of eigenvalues is an indicator of how well the associated eigenvector captures/reconstructs the data in the L_2-norm sense in continuous function space (or the $2-$norm sense in the discretized case). This fact has an important physical interpretation for the velocity and displacement variables in fluid flows and structures respectively. Sorted in descending order of eigenvalue magnitude, the associated eigenfunctions/modes are automatically arranged in decreasing importance w.r.t. to the kinetic energy and strain energy when the variables are velocity and displacement respectively.

Method of Snapshots and Connections with SVD. Sirovich [235] proposed the *method of snapshots* to deal with arbitrarily large field vectors. In cases where $m \gg n$, it usually means that the product $\boldsymbol{R} = \boldsymbol{W}\boldsymbol{W}^T \in \mathbb{R}^{m\times m}$ becomes unwieldy. For realistic problems, computing the decomposition is practically infeasible. An alternative approach to compute identical spatial eigenfunctions/modes is to work with $\tilde{\boldsymbol{R}}$ (eq. 19) and solve the much cheaper eigenvalue problem given by equation (20). This approach is also referred to as the *dual formulation of PCA* in numerical linear algebra [100].

$$\tilde{R} = W^T W \in \mathbb{R}^{n\times n},\ n \ll m \tag{19}$$

$$\tilde{R}\boldsymbol{\Psi} = \boldsymbol{\Psi}\tilde{\boldsymbol{\Lambda}},\ \boldsymbol{\Psi} \in \mathbb{R}^{n\times n},\ \tilde{\boldsymbol{\Lambda}} = \mathrm{diag}(\lambda_1, \ldots, \lambda_n) \tag{20}$$

Eigenvectors of the smaller problem can be used to recover eigenvectors of the original large problem by leveraging the relationship between eigendecomposition of $\boldsymbol{W}\boldsymbol{W}^T$ and $\boldsymbol{W}^T\boldsymbol{W}$ using the expression

$$\begin{aligned} \boldsymbol{\phi}_i &= \frac{1}{\sqrt{\lambda_i}} W\boldsymbol{\psi}_i,\ i = 1, 2, 3, \ldots, n \\ \Leftrightarrow \boldsymbol{\Phi} &= W\boldsymbol{\Psi}\boldsymbol{\Lambda}^{-0.5},\ \boldsymbol{\Phi} \in \mathbb{R}^{m\times n},\ \boldsymbol{\Psi} \in \mathbb{R}^{n\times n} \end{aligned} \tag{21}$$

Singular Value Decomposition (SVD) is a popular rank revealing factorization of any matrix. To see the connection between SVD and the eigendecomposition of $\boldsymbol{W}\boldsymbol{W}^T$ and $\boldsymbol{W}^T\boldsymbol{W}$, consider the SVD of the snapshot/data matrix.

$$\boldsymbol{W} = \boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Psi}^T,\ \boldsymbol{\Phi} \in \mathbb{R}^{m\times m},\ \boldsymbol{\Sigma} \in \mathbb{R}^{m\times n},\ \boldsymbol{\Psi} \in \mathbb{R}^{n\times n}$$

Here $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ are orthonormal matrices containing as columns the left and right singular vectors and $\boldsymbol{\Sigma} = \mathrm{diag}(\sigma_1, \sigma_2, \ldots,\ \sigma_n)$ contains the singular values arranged in descending order. The relationship mentioned above can be seen by the following straightforward manipulations

$$\begin{aligned} \boldsymbol{W}\boldsymbol{W}^T &= \boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Psi}^T\boldsymbol{\Psi}\boldsymbol{\Sigma}\boldsymbol{\Phi}^T = \boldsymbol{\Phi}\boldsymbol{\Sigma}^2\boldsymbol{\Phi}^T \\ \text{or, } \boldsymbol{W}\boldsymbol{W}^T\boldsymbol{\Phi} &= \boldsymbol{\Phi}\boldsymbol{\Sigma}^2 \\ \text{and, } \boldsymbol{W}^T\boldsymbol{W} &= \boldsymbol{\Psi}\boldsymbol{\Sigma}\boldsymbol{\Phi}^T\boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Psi}^T = \boldsymbol{\Psi}\boldsymbol{\Sigma}^2\boldsymbol{\Psi}^T \\ \text{or, } \boldsymbol{W}^T\boldsymbol{W}\boldsymbol{\Psi} &= \boldsymbol{\Psi}\boldsymbol{\Sigma}^2 \end{aligned}$$

The equations above show that $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ are the eigenvectors of $\boldsymbol{W}\boldsymbol{W}^T$ and $\boldsymbol{W}^T\boldsymbol{W}$ respectively. In addition, $\lambda_i = \sigma_i^2$. Therefore in practice, the modes for POD can be obtained by performing the SVD on matrix W instead of the eigendecomposition. Although for large data sets, the method of snapshots is preferred, it is known to be prone to numerical round-off errors. Therefore, the more robust and numerically stable [8] SVD is preferred instead. Another vantage point to clearly see the optimal reconstruction property of the SVD is through the result shown below. Note that this result is a key property of the SVD leveraged in almost every application.

$$\sum_{i=1}^{n} ||\boldsymbol{w}_i - \boldsymbol{\Phi}\boldsymbol{\Phi}^T\boldsymbol{w}_i||_2^2 = \sum_{j=k+1}^{\min(m,n)} \sigma_j^2$$

Summary, Pros and Cons. An attractive feature of the POD is that the modes are orthogonal. In other words, the orthogonal map allows for an explicit transformation from and into the reduced space. The method is relatively simple to use in practice (for compressing high-dimensional data). For experimental data however, noise appears as higher order POD modes [243]. Simply ignoring those modes can remove the noise (under the assumption that it is weaker than the signal). In unsteady cases, the POD modes contain a mix of frequencies which may result in inaccurate predictions. Moreover, as stated above, the modes are sorted based on 2nd order correlations (energy) as opposed to dynamical importance. When considering time-dependent systems, this is undesirable. We will see how the Dynamic Mode Decomposition (DMD) addresses this issue later. Moreover, higher-order correlations are ignored inherently by the mathematical formulation of POD. Another important fact about POD is that even though the reconstruction error is minimized, the resultant ROM for dynamical systems can perform poorly because certain low-energy states can have a large influence on the dynamics yet not contribute substantively to the low-order POD modes [219, 243].

With regard to computation of basis vectors for parametric ROMs, after the cost of running the full order simulation, the SVD-based POD algorithm is the most computationally expensive and memory heavy *offline* operation. The complexity of computing the SVD for an $m \times n$ matrix is $\mathcal{O}[\min(m^2n, mn^2)]$. Although for very large simulations ($\mathcal{O}[10^8]$), the

relative cost of the SVD is negligible when compared to the cost of running the simulation, costs of storing, reading and writing become bottlenecks in practice. This is especially amplified when dealing with time-dependent full order parametric systems.

2.3 Handling Parametric Dependence for Reduced Order Models

For the framework laid out above, we will now look at techniques that enable construction of ROMs at different parameter points. This discussion will organically lead into the topic of non-intrusive parametric ROMs where the system matrices/operators are not accessible. The discussion closely follows that of the survey paper by Benner et. al. [31]. All the techniques will be demonstrated using simple linear systems as a working example. Note that although the focus of this work is on static systems, the presentation is applicable to time-dependent, parametric systems.

2.3.1 Global Basis

Say the full model is evaluated at parameter points $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_n$ from a design of experiments (DoE) to obtain trajectories $W_1 = [\boldsymbol{w}(\boldsymbol{\mu}_1, t_1), \boldsymbol{w}(\boldsymbol{\mu}_1, t_2)], \ldots]$, W_2, $W_3, \ldots,$ W_n. Under the assumption that a rank-k truncation captures sufficient energy according to the RIC criterion, a separate POD basis set can be computed for each $\boldsymbol{\mu}_i$ to yield $\boldsymbol{\Phi}_i \in \mathbb{R}^{m \times k}$, $i = 1, 2, \ldots, n$. The global basis set is obtained by concatenation of the basis sets computed for each parameter point.

$$\boldsymbol{\Phi} = [\boldsymbol{\Phi}_1, \boldsymbol{\Phi}_2, \ldots, \boldsymbol{\Phi}_n]$$

Usage of the term "global" signifies the fact that this set is valid over the entire parameter domain. In other words, once this basis set is created *offline*, it used as is in the *online* phase for projection at an unseen parameter point. In this sense, the modes are applicable 'globally' in the parameter domain. Therefore, favorable performance of the ROM at a new unseen parameter point is directly dependent on the effective sampling of the parameter space in the *offline* phase. The global basis set is often formed by first concatenating followed by a rank-revealing, orthogonal QR/SVD step for two main reasons: 1) to get rid of directions among matrices that are co-linear and 2) for numerical stability (just like

algorithms to compute numerically robust Krylov subspaces).

2.3.2 Interpolation Among Local Bases

As the name suggests, the local approach involves computing basis sets $\boldsymbol{\Phi}_i \in \mathbb{R}^{m \times k},\ i = 1, 2, \ldots, n$ for each $\boldsymbol{\mu}_i$ followed by some form of interpolation as a function of the parameters/design/decision variables. Note that the exact basis set is not as important as the subspace it represents because there may be different basis sets that range the same subspace. All that changes for the ROM is the value of the reduced space coordinates. In practice therefore, Amsallem et. al. [13] realized that the subspace must be interpolated rather than the basis vectors. Such schemes consider subspace matrices as quantities on manifolds [284]. The interpolation is performed on the tangent space (which is locally Euclidean) to the manifold. As such, the interpolation scheme ensures that properties, such as orthogonality or positive definiteness, of the interpolated quantities are preserved.

$$\mathbb{V} : \{\boldsymbol{\Phi}_{\boldsymbol{\mu}_1},\ \boldsymbol{\Phi}_{\boldsymbol{\mu}_2}, \ldots,\ \boldsymbol{\Phi}_{\boldsymbol{\mu}_n}\}$$

$$\boldsymbol{\Phi}(\tilde{\boldsymbol{\mu}}) = \text{Interpolant}(\mathbb{V}, \boldsymbol{\mu})$$

2.3.3 Interpolation Among Reduced System Operators

The methods presented above only discuss techniques to handle parametric dependence in the reduced order basis sets. When in use with a new unseen point in the parameter space, the projection step i.e. matrix multiplications must be performed repeatedly. Techniques that fall under this category obviate the (often expensive) matrix multiplication step by interpolating quantities in the reduced space [11, 14, 80, 187]. The *offline* cost involves performing the projection step. The reduced system quantities are used *online* to perform the interpolation as a function of the parameters. The basic procedure involves *congruence transformations* that make representation of the local basis sets consistent. Depending on whether the basis sets are local or global, the projection step is followed by either element-wise [191] or matrix manifold interpolation [85, 284]. Note that this approach reduces the *online* cost because the interpolation process depends on the reduced space dimension (k) instead of the much larger full order dimensions (m or n). The equations presented below

show how this approach treats parametric variation for linear time-dependent systems. Say we have local basis vectors and system operators for a DoE of parameters $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_n$ denoted as

$$\mathbb{V} : \{\boldsymbol{\Phi}_{\boldsymbol{\mu}_1},\ \boldsymbol{\Phi}_{\boldsymbol{\mu}_2}, \ldots,\ \boldsymbol{\Phi}_{\boldsymbol{\mu}_n}\}$$
$$\mathbb{A} : \{\boldsymbol{A}(\boldsymbol{\mu}_1),\ \boldsymbol{A}(\boldsymbol{\mu}_2), \ldots, \boldsymbol{A}(\boldsymbol{\mu}_n)\}$$

Following a congruence transformation [187], a transformed set of bases is obtained as

$$\tilde{\mathbb{V}} : \{\tilde{\boldsymbol{\Phi}}_{\boldsymbol{\mu}_1},\ \tilde{\boldsymbol{\Phi}}_{\boldsymbol{\mu}_2}, \ldots,\ \tilde{\boldsymbol{\Phi}}_{\boldsymbol{\mu}_n}\}$$
$$\text{and, } \mathbb{A}_r : \{\tilde{\boldsymbol{\Phi}}^T_{\boldsymbol{\mu}_1}\boldsymbol{A}(\boldsymbol{\mu}_1)\tilde{\boldsymbol{\Phi}}_{\boldsymbol{\mu}_1},\ \tilde{\boldsymbol{\Phi}}^T_{\boldsymbol{\mu}_2}\boldsymbol{A}(\boldsymbol{\mu}_2)\tilde{\boldsymbol{\Phi}}_{\boldsymbol{\mu}_2}, \ldots, \tilde{\boldsymbol{\Phi}}^T_{\boldsymbol{\mu}_n}\boldsymbol{A}(\boldsymbol{\mu}_n)\tilde{\boldsymbol{\Phi}}_{\boldsymbol{\mu}_n}\}$$
$$\Leftrightarrow \mathbb{A}_r : \{\tilde{\boldsymbol{A}}_r(\boldsymbol{\mu}_1),\ \tilde{\boldsymbol{A}}_r(\boldsymbol{\mu}_2), \ldots, \tilde{\boldsymbol{A}}_r(\boldsymbol{\mu}_n)\}$$
$$\tilde{\boldsymbol{A}}_r(\tilde{\boldsymbol{\mu}}) = \text{Interpolant}(\mathbb{A}_r, \boldsymbol{\mu})$$

where $\mathbb{A}_r$ represents the set of reduced system matrices. For a new parameter point $\hat{\boldsymbol{\mu}}$, the elements of $\mathbb{A}_r$ are used to train an interpolating function that depends on $\boldsymbol{\mu}$ to yield a new reduced operator $\tilde{\boldsymbol{A}}_r(\hat{\boldsymbol{\mu}})$.

2.3.4 Treatment of Parametric Dependence with Non-Intrusive ROMs

Notice that all the aforementioned interpolation approaches work exclusively with intrusive methods. This work focuses on situations where the system operators are unavailable. The state-of-art for handling parametric dependence in such cases is interpolation or regression among coordinates in the latent or reduced space. In other words, once a lower dimensional representation $\boldsymbol{w}_r^i \in \mathbb{R}^k,\ i = 1, 2, \ldots$ of each $\boldsymbol{w}_i \in \mathbb{R}^m,\ i = 1, 2, \ldots$ is obtained by projecting onto the basis vectors, the k-dimensional vector $\boldsymbol{w}_r^i$ is treated as a set of scalars that varies with the design/decision variables or parameters. Other approaches exist where the ROM methodology is formulated as an operator inference problem [191]. Once the system matrices are inferred, the interpolation among reduced order matrices approach is used to obtain the ROM at a new parameter point. Recent work by Renganathan et. al. has successfully leveraged *Koopman theory* to devise an approach that exploits knowledge of the governing equations [213] in conjunction with the solutions to create parametric ROMs for steady

systems. Their work also interpolates among reduced symmetric positive definite matrices to obtain an approximation of the reduced system matrix at unseen parameter points.

2.3.5 Closing Remarks and Summary

Depending on whether system operators are available or not, various approaches to obtain ROMs at varying parameter locations were discussed in this section. As with any surrogate modeling method, for the parametric reduced order model to perform with sufficient accuracy, a good sampling of the parameter space must be ensured. For low-dimensional decision variable/parameter spaces ($d \ll 10$), a dense sampling may be feasible. But as the dimensions increase beyond a handful in number ($d \sim \mathcal{O}(10)$) the *curse of dimensionality* takes over and renders the *offline* cost intractable and/or results in a complex ROM. Moreover, regression/interpolation in the lower dimensional coordinates as a function of $\boldsymbol{\mu} \in \mathbb{R}^d$ may also become challenging as the number of parameters increases. As stated in chapter 1, this dissertation is concerned with non-intrusive parametric ROMs for practical problems faced with challenges regarding a large number of inputs and high offline cost. Now that a brief overview of intrusive methods has been given, we are prepared to take an in-depth look at various nonintrusive ROM methods, their developments through time to identify pros, cons and gaps. We will make note of important observations as we proceed to motivate the research formulation and choice of the candidate ROM method to address the challenges identified in chapter 1.

2.4 Non-Intrusive Reduced Order Modeling Methods

The framework for ROMs briefly presented above requires knowledge of the discrete system operators to perform the reduction. We presented reasons to argue (in chapter 1) that this requirement is too restrictive and seldom practical in real world applications. It is on this premise that we focus on non-intrusive parametric ROMs in this work. Methods that fall under this category are also called *equation-free* ROMs. Before proceeding, we re-state some challenges that characterize the traits of typical practical problems that the ROM method should fulfill: 1) must handle parameter dimensions $d \gg 10$, 2) must work for high-dimensional field outputs ($\mathcal{O}(10^{6-7})$) i.e. large datasets, and 3) must work without

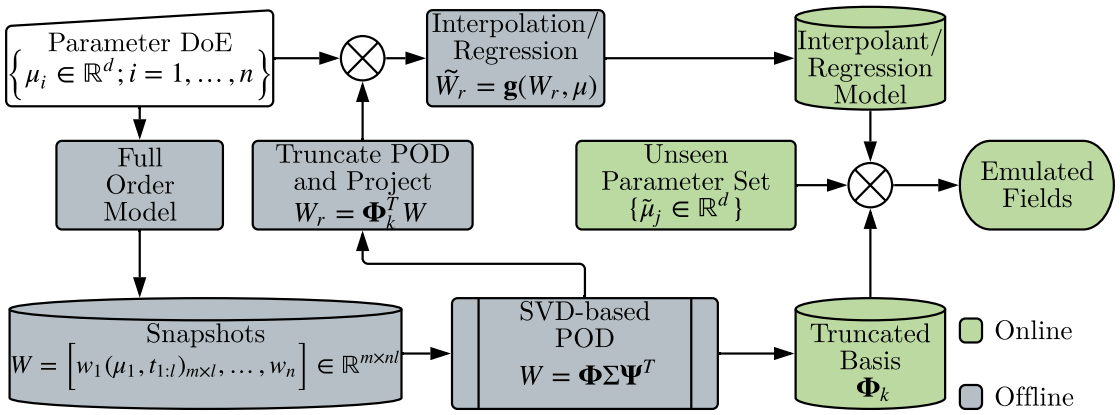


Figure 13: Offline-Online Decomposition in POD with Interpolation

requiring access to the system operators i.e. must be nonintrusive.

2.4.1 POD with Interpolation or Least-Squares in Latent Space

This set of methods is arguably the most popular among practitioners in literature. The basic procedure utilizes the POD method on a set of snapshots (full-order solutions at different parameter locations) to compress and obtain a low-dimensional basis set. The training data is then projected onto the reduced space to obtain coordinates of the high-dimensional system in the latent space. Encoding the parametric variation is achieved by creating a map between the parameter space and coordinates in the latent space. Interpolation and/or regression techniques from supervised machine-learning are usually utilized to learn this map. Figure 13 shows the general process and the *offline-online* decomposition for the method.

In this set of approaches, the *offline* phase consists of evaluating the full order model at a set of predetermined points in the parameter space as specified by the DoE. Once snapshot data is collected, the POD method is used to reduce the dimensionality and find a basis set. The full order solutions are then projected onto the subspace spanned by the range of $\mathbf{\Phi}_k$. Each latent space coordinate is treated as a scalar that is a function of the parameters. The DoE along with the reduced coordinates are used to train scalar surrogate models to serve as maps between design/decision variables and lower dimensional coordinates. At a new unseen point in parameter space, a linear combination of the truncated basis sets and the predicted latent space coordinates are used to emulate the field.

Table 2: Studies on Dimension Reduction and Interpolation-based Parametric ROMs for Academic and Engineering Problems

Problem Description	DR Method	Regression Method	d	m	n	k	Ref.
2D Channel Flow	POD	Cubic Spline	1	-	22	3	[161]
NACA 0012 Airfoil	POD	Cubic Spline	2	2048	231	$\mathcal{O}(10)$	[50]
1D Modified Burgers Eq.	POD	RBF	2	200	81	5	[20]
1D Modified Burgers Eq.	POD	RBF	3	200	49	≤ 10	[20]
2D Modified Burgers Eq.	POD	RBF	5	2500	200	13	[20]
Propulsion System	EM-PCA	Neural Net	6	947	500	4	[141]
Flow Around Wing-Fuselage	POD	Neural Net	5	2.88e5	10	10	[189]
1D Burgers Eq.	POD	RBF	1	100	20	≤ 10	[21]
2D Modified Burgers Eq.	POD	RBF	3	1600	6.25e4	≤ 10	[21]
NACA 64A010 Airfoil	Isomap	RBF	2	400	30	2	[98]
NACA 64A010 Airfoil	POD	RBF	2	400	30	30	[98]
LANN Wing	Isomap	RBF	2	3901	25	2	[98]
LANN Wing	POD	RBF	2	3901	25	25	[98]
2-Stage Turbine	POD	RBF	4	-	$\mathcal{O}(100)$	$\sim$6	[127]
CRM High-Lift	POD	Kriging	1	2.19e7	10	≤ 10	[97]
CRM Wing-Body-Tail	POD	Kriging	2	2.00e7	85	$\mathcal{O}(10)$	[97]
ONERA 7AD Rotor	POD	Kriging	2	-	55	$\mathcal{O}(10)$	[97]
2D Composite Panel	POD	Polynomials	5	3000	3921	$\mathcal{O}(100)$	[163]
NACA 0012 Airfoil	POD	Cubic Spline	1	13490	100	$\mathcal{O}(10)$	[147]
Topology Optimization	POD	Neural Net	4	3200	400	80	[253]
2D Fluid Mixing	POD	RBF	1	28000	1750	$\mathcal{O}(10)$	[268]
2D Flow Around Cylinder	POD	RBF	1	1000	3213	$\mathcal{O}(10)$	[268]
NACA 0012 Airfoil	POD	RBF	1	75000	30	$\mathcal{O}(10)$	[66]
Car Aerodynamics	POD	Kriging	3	1.15e8	$\mathcal{O}(10)$	$\mathcal{O}(10)$	[37]
2D Unsteady Heat Eq	POD	RBF	1	1600	25	$\sim$10	[65]
2D Lid Driven Cavity	POD	RBF	2	1600	30	$\mathcal{O}(10)$	[65]
1D Nonlinear Poisson	POD	Cubic Splines, Neural Nets	3	100	$\mathcal{O}(100)$	5	[117]
2D Nonlinear Poisson	POD	Cubic Splines, Neural Nets	3	2792	$\mathcal{O}(100)$	35	[117]
2D Lid Driven Cavity	POD	Cubic Splines, Neural Nets	3	-	$\mathcal{O}(100)$	$\mathcal{O}(10)$	[117]
1D Burgers Eq	POD	GPR	1	201	2.7e4	7	[109]
2D Flow Around Cylinder	POD	GPR	1	3899	1326	36	[109]
Equilibrium Paths of a Truss	POD	GPR	2	476	10100	-	[109]
1D Heated Rod	POD	Several	1	-	-	-	[242]
2D Composite Panel	POD	Several	5	3921	3000	176	[242]
Airfoil	POD	Several	2	9027	127	10	[242]

A few of the well-known earliest uses of such approaches were by Ly et. al. [161] and Bui-Tanh et. al. [50]. In both these papers, the authors use cubic splines to create interpolants for the coefficients as functions of the parameters and demonstrate accurate field emulation at points not used in the training process. Some authors [54] used this approach in the context of inverse design with incomplete data in conjunction with the gappy POD method [91]. Others [20, 21, 65, 257, 268, 269] have relied on radial basis functions (RBFs) for the reduced space coordinate surrogates. Mainini et. al. [163] used the POD method along with response surfaces and self-organizing maps to enable real-time structural capability assessment. Recently, researchers [117, 253] have experimented with usage of artificial neural networks for the reduced coordinates. In related work by Xiao et. al. [267], sparse grids and Taylor series expansions were used to predict coefficients with time as the only parameter. Authors [37, 97, 109, 257] have also used Gaussian processes for capturing parametric dependence of the latent space coordinates. Table 2 lists details of some important studies that deal with construction of ROMs using a combination of dimension reduction (mostly POD) and interpolation. This chapter ends with a discussion of the common traits that can be observed in all these studies to motivate the effort in this work.

Another POD-based approach leverages the ability to query the governing equations in the residual form to create reduced order models. Specifically, once the POD is performed on a set of snapshots collected by densely sampling the parameter space, the approach seeks solutions for new design/decision variable points by restricting the search space for the residual equations in the subspace spanned by the truncated rank-k POD modes thereby reducing the effective dimensionality. The interpolation step is completely bypassed because the governing equations themselves are being solved albeit in a lower dimensional space. Say the full order model is expressed in the residual form,

$$\boldsymbol{r}(\boldsymbol{w}, \boldsymbol{\mu}) = 0,\ \boldsymbol{w} \in \mathbb{R}^m,\ \boldsymbol{r} : \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R}^m, \text{and},\ \boldsymbol{\mu} \in \mathbb{R}^d$$

and $\boldsymbol{\Phi}_k \in \mathbb{R}^{m \times k},\ k \ll m$ represents the truncated POD basis obtained using solutions at a set of parameter points. Using the ansatz $\boldsymbol{w} = \boldsymbol{\Phi}_k \boldsymbol{w}_r$, this approach solves the following

least-squares problem to evaluate the solution at a new parameter point $\tilde{\boldsymbol{\mu}}$:

$$\min_{\boldsymbol{w}_r} \boldsymbol{r}(\boldsymbol{\Phi}_k \boldsymbol{w}_r, \tilde{\boldsymbol{\mu}})$$

An advantage is that the method is relatively more physics-based than the interpolation approach because the actual residual equations are being solved for, albeit in a subspace of considerably smaller size. Figure 14 shows the *offline-online* decomposition for this method. Note that the offline cost is identical to that of the approach with interpolation. However,

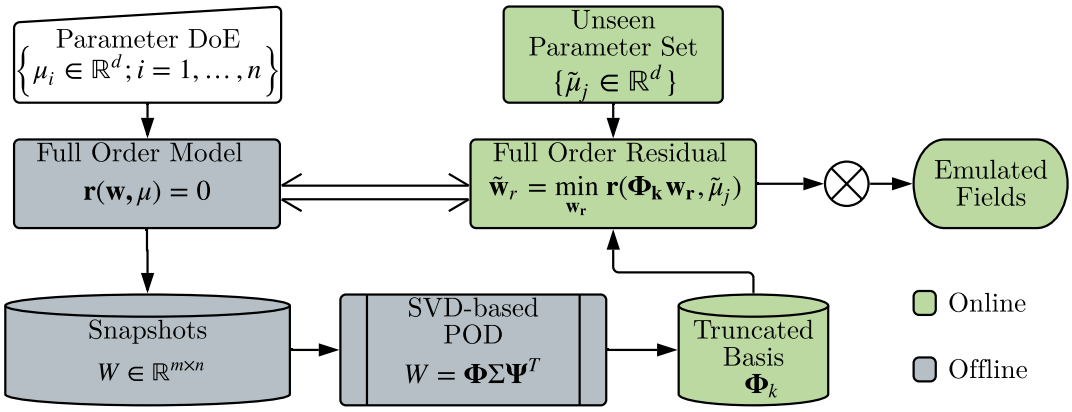


Figure 14: Offline-Online Decomposition in POD with Least-Squares

the online cost is different. On observing closely, even though the solution search space is smaller. all the terms in the nonlinear residual equations must be evaluated i.e. there is no reduction in the system of equations even though the degrees of freedom are smaller. Therefore, if m is large and the residual is expensive to evaluate, the *online* cost does not enjoy a significant reduction in time complexity. Perhaps the earliest use of this method in the context of fluids was put forth by LeGresley et. al. [143]. A variety of applications in different contexts along with improvements was suggested with time [282, 285, 286]. Several issues, some shared and others not with the interpolation-based methods require attention, when employing this method in practice: 1) there exists no guarantee on the quality or existence of a solution to the least-squares problem in the lower-dimensional POD basis (although it usually works in practice), 2) quality of the solution for an untried parameter point depends on the sampling of snapshots in the design/decision variable space, and 3) even though computational savings have been reported in practice, evaluation of the residual

equations still depends on the full order model dimension (m). Therefore, truly intrusive methods that allow the projection of the residual terms onto the POD basis offer better computational savings.

2.4.2 Data-Driven Operator Inference

Proposed by Peherstorfer et. al. [191], this non-intrusive projection-based method uses initial conditions and trajectories of the state variables and outputs to project and infer the reduced operators directly from the black-box simulation. Its mathematical formulation is applicable to time-dependent parametric systems whose underlying governing equations contain at most low-order polynomial non-linear terms. Therefore, the method relies on knowing the analytical form of the governing equation preferably along with the knowledge of the analytical form of parametric dependence. A salient and desirable feature of their work is a proof that ensures convergence of the inferred reduced operators to the projected operators that would be obtained if the operators were accessible in the *Frobenius*-norm if sufficiently rich trajectory data were available.

What follows is a demonstration of their method on a quadratic-in-state parametric problem using the notation used above. Although the focus of this presentation is computation of the state only, the original work considers computation of both states and outputs that depend on the state through a linear map. Consider a discretized system of parametric ODEs

$$\begin{aligned} \frac{d\boldsymbol{w}(t,\boldsymbol{\mu})}{dt} &= \boldsymbol{A}(\boldsymbol{\mu})\boldsymbol{w}(t,\boldsymbol{\mu}) + \boldsymbol{F}(\boldsymbol{\mu})\boldsymbol{w}(t,\boldsymbol{\mu})^2 \\ \text{and, } \boldsymbol{w}(0,\boldsymbol{\mu}) &= \boldsymbol{w}_0(\boldsymbol{\mu}) \\ \text{where, } \boldsymbol{A}(\boldsymbol{\mu}) \in \mathbb{R}^{m\times m},\ \boldsymbol{w}(t,\boldsymbol{\mu}) &\in \mathbb{R}^m, \text{ and } \boldsymbol{F}(\boldsymbol{\mu}) \in \mathbb{R}^{m\times s} \end{aligned}$$

The terms $\boldsymbol{A}(\boldsymbol{\mu}),\ \boldsymbol{F}(\boldsymbol{\mu})$ are the linear and quadratic terms for the ODE system at parameter point $\boldsymbol{\mu} \in \mathbb{R}^d$ with full order size m. The term s denotes the number of terms that arise due to representation of the nonlinearity. As such, the vector $\boldsymbol{w}(t,\boldsymbol{\mu})^2$ represents all the products of two components of $\boldsymbol{w}(t,\boldsymbol{\mu})$ and therefore $s = m(m+1)/2$.

The inference problem assumes availability of the trajectory data at a DoE of design/decision variables $\{\boldsymbol{\mu}_i; \boldsymbol{\mu} \in \mathbb{R}^d\}_{i=1}^n$ to create the reduced order model. Instead of inferring the full order model (which is large in dimension m), a global POD basis is used to first project the data and solve the inference problem in the reduced space at each parameter point in the DoE. The global basis set is constructed as mentioned above (see section 2.3.1) using the following trajectory set,

$$\mathbb{W} = [\boldsymbol{w}_0(\boldsymbol{\mu}_1),\ \boldsymbol{W}(\boldsymbol{\mu}_1), \ldots, \boldsymbol{w}_0(\boldsymbol{\mu}_m),\ \boldsymbol{W}(\boldsymbol{\mu}_m)] \in \mathbb{R}^{m\times(l+1)n}$$

where $\boldsymbol{W}(\boldsymbol{\mu}_i) \in \mathbb{R}^{m\times l}$ is a matrix containing evolution of state at design/decision variable value $\boldsymbol{\mu}_i$ and l is the number of time steps the state is advanced for at each parameter point. The POD is applied to this entire data set to obtain a global rank-k truncated basis set

$$\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \ldots, \boldsymbol{\phi}_k] \in \mathbb{R}^{m\times k}$$

The inference procedure proceeds by projecting the data matrix and approximate time derivatives of the state (calculated for instance using finite-differences) onto the global basis set for each parameter point in the DoE.

$$\tilde{\boldsymbol{W}}(\boldsymbol{\mu}_i) = \boldsymbol{\Phi}^T \boldsymbol{W}(\boldsymbol{\mu}_i)$$
$$\tilde{\dot{\boldsymbol{W}}}(\boldsymbol{\mu}_i) = \boldsymbol{\Phi}^T \dot{\boldsymbol{W}}(\boldsymbol{\mu}_i)$$
$$\text{where, } i = 1, 2, \ldots, n$$

$\dot{\boldsymbol{W}}(\boldsymbol{\mu}_i) = [\dot{\boldsymbol{w}}_0(\boldsymbol{\mu}_i), \dot{\boldsymbol{w}}(t_1, \boldsymbol{\mu}_i), \ldots, \dot{\boldsymbol{w}}(t_l, \boldsymbol{\mu}_i)]$ is the matrix containing approximate time derivatives of the state. Then the reduced operator inference is found by solving the minimization problem given by

$$\min_{\tilde{\boldsymbol{A}}\in\mathbb{R}^{k\times k}, \tilde{\boldsymbol{F}}\in\mathbb{R}^{k\times s}} ||\tilde{\dot{\boldsymbol{W}}}(\boldsymbol{\mu}_i) - \tilde{\boldsymbol{A}}(\boldsymbol{\mu}_i)\tilde{\boldsymbol{W}}(\boldsymbol{\mu}_i) - \tilde{\boldsymbol{F}}(\boldsymbol{\mu}_i)\tilde{\boldsymbol{W}}(\boldsymbol{\mu}_i)^2||_2^2$$

Note that n such independent least-squares problems must be solved, one for each parameter point. However, since dimensions of the inferred operators $k \ll m$, solving the least-squares problem is feasible for reasonably moderate values of n (size of the DoE). The resulting database of ROMs thus obtained can be exploited along with interpolation to obtain reduced

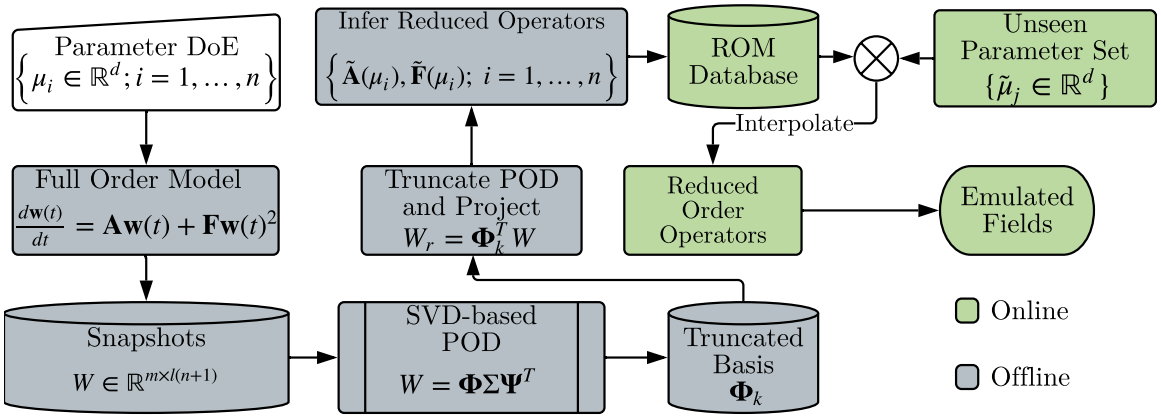


Figure 15: Offline-Online Decomposition in Data-Driven Operator Inference

matrices for new parameter points without recourse to the inference problem. In their work, the authors used element-wise spline interpolation to obtain the reduced order operators at unseen parameter points. The *offline* cost for solving the optimization problem n times is bounded by $\mathcal{O}[n \times kl(k+s)^3]$. The *online* cost involves the interpolation step complexity and the solution of the reduced order system dependent on k and l (independent of full order dimension m). Figure 15 shows the offline-online decomposition for this method.

In practice, the method successfully predicted across behavior regimes (i.e. stable, unstable, limit-cycle) at unseen parameter values. However, the examples had at most two independent parameters. Another important requirement for the method is knowledge of the governing equation form to formulate the inference problem.

2.4.3 Dynamic (Koopman) Mode Decomposition

Originally developed for applications in the fluid dynamics community by Schmid et. al. [230] to decompose time-resolved data into spatio-temporal modes each having a unique oscillation frequency with an associated decay/growth rate, the Dynamic Mode Decomposition (DMD) (and more recently its generalization to nonlinear systems through *Koopman theory* [133]) has emerged as a purely data-driven, *equation free* method for explaining time series data, constructing reduced order models, and as a diagnostic tool providing insight into the behavior of a system. The DMD has firm roots in numerical linear algebra because of its dependence on the eigendecomposition of a best-fit linear operator to approximate

(while attempting to diagonalize) dynamics of a system as captured by data. Another desirable and promising feature of the Koopman-based DMD is its ability to also approximate the dynamics of nonlinear systems under appropriate conditions [47, 172, 173, 179, 220]. DMD has also been touted as a combination of the POD and the discrete Fourier transform [243] having a wide variety of application areas including neuroscience [45], robotics [34], financial trading [165], video processing [108] etc.

More importantly, for time-dependent systems DMD's use for constructing reduced order models comes with several benefits: 1) the method gives importance to the dynamics of evolution rather than energy (as in POD) and 2) an organic consideration of time in the mathematical formulation (which is objectively unclear in purely POD-based methods). Many variants [18, 123, 137, 148, 182, 263, 266] of the first basic algorithm [229] have been developed claiming improvements under some assumptions on data with regard to accuracy, efficiency, and robustness in the computation of the eigendecomposition. Despite immensely successful applications in a wide gamut of disciplines, the DMD lacks mathematically rigorous certificates of accuracy. Recent work has made significant progress towards addressing this limitation [82, 139].

While it has been employed as a non-intrusive ROM enabler for systems without parametric dependence [9, 40–42, 123, 131, 182, 245, 264, 280] its parametric extension does not exist as noted by several researchers [15, 157, 191]. The following discussion briefly describes the basic algorithm, its development as a ROM, and applications to problems with nonlinear dynamics. The notation and exposition follow the textbook by Kutz et. al. [136].

As mentioned before, DMD can be thought of as a combination of dimensionality reduction technique such as the POD along with a Fourier transform in time. At its core, it is a numerical technique leveraging linear algebra to compute a decomposition/factorization that explains data. In that sense, it is regressive in nature just as methods in machine learning are, except that the spatial and temporal relationships are retained. The DMD starts from a nonlinear dynamical system arising from the spatial discretization of a set of

coupled parametric PDEs

$$\frac{d\boldsymbol{w}}{dt} = \boldsymbol{f}(\boldsymbol{w}, t, \boldsymbol{\mu})$$

where $\boldsymbol{w} \in \mathbb{R}^m$ is the state, $t \in \mathbb{R}_+$ denotes time, $\boldsymbol{\mu} \in \mathbb{R}^d$ represents the parameters, and function $\boldsymbol{f} : \mathbb{R}^m \times \mathbb{R}^d \times \mathbb{R}_+ \to \mathbb{R}^m$ contains the nonlinear dynamics. As mentioned several times before, $m \gg 1$ i.e. the system is high-dimensional and the number of dimensions directly depends on the granularity of spatial discretization. The data is collected from a process that evolves and samples the system at every Δt using the time-discrete map

$$\boldsymbol{w}_{q+1} = \boldsymbol{F}(\boldsymbol{w}_q), \quad q = 1, 2, \ldots, n$$

where $\boldsymbol{w}_q = \boldsymbol{w}(q\Delta t)$ and a total of n state measurements are collected. Along with the initial conditions, $\boldsymbol{w}(t = 0) = \boldsymbol{w}_0$ and the governing equations, a well-posed problem is specified. The DMD architecture adopts the *equation free* perspective, where $\boldsymbol{f}$ is unknown and uses data exclusively to explain the dynamics and predict future state by constructing an approximate locally tangent linear dynamical system or a proxy given by

$$\frac{d\boldsymbol{w}}{dt} = \mathcal{A}\boldsymbol{w}, \quad \mathcal{A} \in \mathbb{R}^{m \times m} \tag{22}$$

$$\text{or, } \boldsymbol{w}(t) = \sum_{i=1}^{m} \boldsymbol{\phi}_i \exp(\omega_i t) b_i = \boldsymbol{\Phi} \exp(\boldsymbol{\Omega} \boldsymbol{t}) \boldsymbol{b} \tag{23}$$

where $\boldsymbol{\Phi} \in \mathbb{R}^{m \times m}$ contains the eigenvectors, $\boldsymbol{\Omega} = \text{diag}(\omega_1, \omega_2, \ldots, \omega_m) \in \mathbb{R}^{m \times m}$ contains the eigenvalues, and $\boldsymbol{b} \in \mathbb{R}^m$ denotes the coordinates of $\boldsymbol{w}_0$ in the eigenbasis. The discrete counterpart for the time continuous system in eq. 22 is

$$\boldsymbol{w}_{q+1} = \boldsymbol{A}\boldsymbol{w}_q \tag{24}$$

$$\boldsymbol{w}_q = \sum_{i=1}^{m} \boldsymbol{\phi}_i \lambda_i^q b_i = \boldsymbol{\Phi} \boldsymbol{\Lambda}^q \boldsymbol{b} \tag{25}$$

where $\boldsymbol{A} = \exp(\mathcal{A}\Delta t)$. The DMD attempts to find matrix $\boldsymbol{A}$ that optimally approximates trajectory $\boldsymbol{w}_q,\ q = 1, 2, \ldots, n$ in a least-squares sense such that the quantity $||\boldsymbol{w}_{q+1} - \boldsymbol{A}\boldsymbol{w}_q||_2$ is minimized for all values of q by solving,

$$\underset{\boldsymbol{A}}{\text{minimize}} \; ||\boldsymbol{W}' - \boldsymbol{A}\boldsymbol{W}||_F \tag{26}$$

where $||.||_F$ is the standard Frobenius norm and

$$\boldsymbol{W} = \begin{bmatrix} | & | & & | \\ \boldsymbol{w}_1 & \boldsymbol{w}_2 & \dots & \boldsymbol{w}_{n-1} \\ | & | & & | \end{bmatrix}, \text{ and } \boldsymbol{W'} = \begin{bmatrix} | & | & & | \\ \boldsymbol{w}_2 & \boldsymbol{w}_3 & \dots & \boldsymbol{w}_{n} \\ | & | & & | \end{bmatrix} \tag{27}$$

resulting in the approximation[3],

$$\boldsymbol{W'} \approx \boldsymbol{A}\boldsymbol{W}, \text{ where } \boldsymbol{A} = \boldsymbol{W'}\boldsymbol{W}^{\dagger}$$

The initially proposed algorithm relied on the popular Arnoldi algorithm to find $\boldsymbol{A}$ owing to the similarity between Krylov subspaces and the sequence created by eq. (24) and required a constant Δt. These assumptions were relaxed subsequently by Tu et. al. [137] who proposed the following broad definition for data not necessarily emanating from a dynamical system or time series data:

Definition - Dynamic Mode Decomposition (Tu et. al. [137]). *For data given as,*

$$\boldsymbol{X} = \begin{bmatrix} | & | & & | \\ \boldsymbol{x}_1 & \boldsymbol{x}_2 & \dots & \boldsymbol{x}_{n} \\ | & | & & | \end{bmatrix}, \qquad \boldsymbol{Y} = \begin{bmatrix} | & | & & | \\ \boldsymbol{y}_1 & \boldsymbol{y}_2 & \dots & \boldsymbol{y}_{n} \\ | & | & & | \end{bmatrix} \tag{28}$$

define the operator

$$\boldsymbol{A} = \boldsymbol{Y}\boldsymbol{X}^{\dagger} \tag{29}$$

where † represents the psuedo-inverse. The Dynamic Mode Decomposition of the pair $(\boldsymbol{X}, \boldsymbol{Y})$ is given by the eigendecomposition of $\boldsymbol{A}$ i.e. the DMD modes and eigenvalues are the eigenvalues and eigenvectors of $\boldsymbol{A}$.

Note that because $m \gg n$, computing $\boldsymbol{A}$ directly can easily become intractable. DMD considers a reduced k-rank representation of $\boldsymbol{A}$ denoted by $\tilde{\boldsymbol{A}} \in \mathbb{R}^{k \times k}$ using projection onto POD modes. The basic DMD algorithm due to [137] proceeds as detailed in algorithm 1. The formula given by $\boldsymbol{\Phi} = \boldsymbol{W'}\boldsymbol{V}\boldsymbol{\Sigma}^{-1}\boldsymbol{Q}$ is referred to as the *exact* DMD while $\boldsymbol{\Phi} = \boldsymbol{U}\boldsymbol{Q}$ is the

[3] † denotes the Moore-Penrose pseudo-inverse.

Algorithm 1: Basic DMD Algorithm

Input: $\boldsymbol{W} \in \mathbb{R}^{m \times n}$, k

Output: $\boldsymbol{\Phi} \in \mathbb{R}^{m \times k}$, $\boldsymbol{\Lambda} \in \mathbb{R}^{k \times k}$

Create $\boldsymbol{W}'$ using $\boldsymbol{W}$ per eq. 27

$\boldsymbol{W} := \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$ // SVD for dimension reduction

$\boldsymbol{U} := \boldsymbol{U}(:, 1:k)$ // choose k left singular vectors for projection

$\tilde{\boldsymbol{A}} := \boldsymbol{U}^*\boldsymbol{A}\boldsymbol{U} = \boldsymbol{U}^*\boldsymbol{W}'\boldsymbol{V}\boldsymbol{\Sigma}^{-1}$ // since $\boldsymbol{A} = \boldsymbol{W}'\boldsymbol{V}\boldsymbol{\Sigma}^{-1}$

/* $\tilde{\boldsymbol{A}}$ is a k-dimensional linear model such that $\tilde{\boldsymbol{w}}_{q+1} = \tilde{\boldsymbol{A}}\tilde{\boldsymbol{w}}_q$, $\boldsymbol{w}_q = \boldsymbol{U}\tilde{\boldsymbol{w}}_q$ */

Solve $\tilde{\boldsymbol{A}}\boldsymbol{Q} = \boldsymbol{Q}\boldsymbol{\Lambda}$ // eigendecomposition of a $k \times k$ system

/* Recontruct the DMD Modes */

if *exact* **then**

  $\boldsymbol{\Phi} = \boldsymbol{W}'\boldsymbol{V}\boldsymbol{\Sigma}^{-1}\boldsymbol{Q}$

else if *projected* **then**

  $\boldsymbol{\Phi} = \boldsymbol{U}\boldsymbol{Q}$

Return $\boldsymbol{\Phi}$ and $\boldsymbol{\Lambda}$

so-called *projected* DMD. Both these converge as the ranges of $\boldsymbol{W}$ and $\boldsymbol{W}'$ become identical. The approximate low-rank state for future times can be computed with the eigenvalues and the DMD modes using

$$\boldsymbol{w}(t) \approx \sum_{i=1}^{k} \boldsymbol{\phi}_i \exp(\omega_i t) b_i = \boldsymbol{\Phi}\exp(\boldsymbol{\Omega}t)\boldsymbol{b} \tag{30}$$

where $\omega_k = \log \lambda_i / \Delta t$, and the lower dimensional initial conditions $\boldsymbol{b}$ can be obtained using $\boldsymbol{b} = \boldsymbol{\Phi}^{\dagger}\boldsymbol{w}_0$.

Handling Nonlinearities - Connections of DMD with Koopman Theory. Pioneered by Bernard Koopman in the 1930s [133] originally for Hamiltonian systems, the Koopman operator trades nonlinear finite-dimensional dynamics for linear infinite-dimensional dynamics. The definition (see below) posits the existence of an infinite dimensional linear operator that acts on nonlinear observables (a set of nonlinear functions of the state). While the definition does not provide a procedure for selecting observables, an appropriate choice is critical for successful application. The following definition applies to continuous (in time) systems.

Definition: Koopman Operator (from [136]) *Consider a continuous-time dynamical system given by,*

$$\frac{d\boldsymbol{w}}{dt} = \boldsymbol{f}(\boldsymbol{w}) \tag{31}$$

where $\boldsymbol{w} \in \mathcal{M}$ is the state on a smooth m-dimensional manifold $\mathcal{M}$. The Koopman operator $\mathcal{K}$ is an infinite-dimensional linear operator that acts on all observable functions $g : \mathcal{M} \to \mathbb{C}$ such that:

$$\mathcal{K}g(\boldsymbol{w}) = g(\boldsymbol{f}(\boldsymbol{w})) \tag{32}$$

The discrete time dynamical system (see eq. (33)) induced from eq. (31) gives rise to an analogous discrete-time Koopman operator $\mathcal{K}_t$ acting on the observable g, resulting in a dynamical system on g instead of $\boldsymbol{w}$.

$$\boldsymbol{w}_{q+1} = \boldsymbol{F}_t(\boldsymbol{w}_q) \tag{33}$$

$$\mathcal{K}_t g = g \circ \boldsymbol{F}_t \tag{34}$$

$$\implies \mathcal{K}_t g(\boldsymbol{w}_q) = g(\boldsymbol{F}_t(\boldsymbol{w}_q)) = g(\boldsymbol{w}_{q+1}) \tag{35}$$

Equation (35) shows that due to the Koopman theory, the nonlinear evolution of the state gets translated into a linear evolution of the observables. The vast rich literature for linear systems can be leveraged to analyze the resulting linear dynamical system. Consider the spectral decomposition (eigendecomposition in the finite dimensional case) of the Koopman operator,

$$\mathcal{K}\psi_k = \lambda_k \psi_k$$

where the Koopman eigenfunctions ψ_k define the set of coordinates to advance with a linear system representation. Therefore, the evolution of a vector observable $\boldsymbol{g}$ can be expanded using the eigenfunction expansion as

$$\boldsymbol{g}(\boldsymbol{w}) = \begin{bmatrix} g_1(\boldsymbol{w}) \\ g_2(\boldsymbol{w}) \\ \vdots \\ g_r(\boldsymbol{w}) \end{bmatrix} = \sum_{k=1}^{\infty} \psi_k(\boldsymbol{w}) \begin{bmatrix} \langle g_1, \psi_k \rangle \\ \langle g_2, \psi_k \rangle \\ \vdots \\ \langle g_r, \psi_k \rangle \end{bmatrix} \boldsymbol{v}_k = \sum_{k=1}^{\infty} \psi_k \boldsymbol{v}_k \tag{36}$$

The promise offered by Koopman theory is the equivalence between any nonlinear system represented by $\boldsymbol{f}$ and its corresponding linear system represented by $\mathcal{K}$. The DMD is used to approximate Koopman eigenpairs $(\lambda_k, \boldsymbol{v}_k)$. With a good set of observables that can be linked to the Koopman decomposition, one can evolve the system using eq. (36) such that

$$\mathcal{K}\boldsymbol{g}(\boldsymbol{w}) = \mathcal{K}\sum_{k=1}^{\infty}\psi_k\boldsymbol{v}_k = \sum_{k=1}^{\infty}\mathcal{K}\psi_k\boldsymbol{v}_k = \sum_{k=1}^{\infty}\lambda_k\psi_k\boldsymbol{v}_k \tag{37}$$

In fact, if the observable functions lie in an invariant subspace spanned by the Koopman eigenfunctions, a discrete finite dimensional linear operator $\boldsymbol{K}$ that can be used to evolve $\boldsymbol{g}$ is induced [46]. In other words, the eigenfunctions provide a coordinate system intrinsic to the system along which the behavior appears linear [243]. Figure 16 shows an illustration of this advantage due to Koopman theory. Critical to the success of the Koopman promise is the selection of appropriate observables. A methodical procedure for selection of the observables is absent in literature. As such, it is an open problem [138].

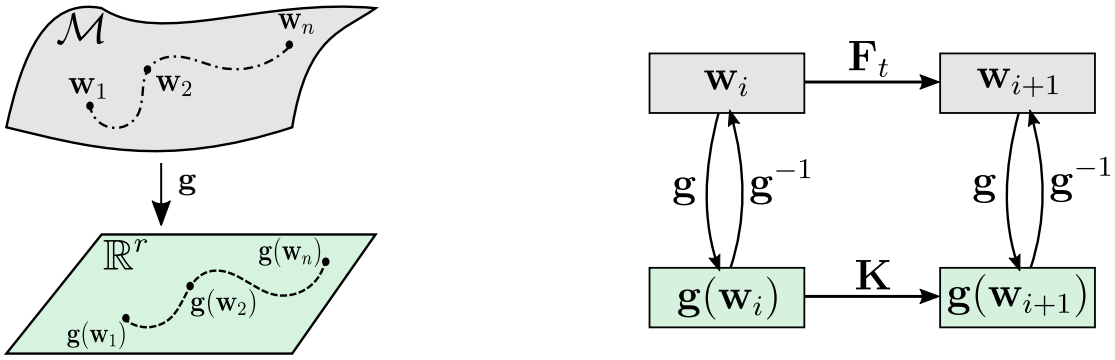


Figure 16: Schematic Illustrating Restriction of the Observables to the Koopman Subspace (adapted from [46])

As mentioned earlier, the DMD may be used to compute the Koopman eigenvalues and modes from data under appropriate choice of observables. Before applying DMD, r observable functions $g_i : \mathcal{M} \to \mathbb{C}$ must be chosen to construct $\boldsymbol{g} = [g_1, g_2, \ldots, g_r]^T$ followed by construction of matrices $\boldsymbol{Y} = \boldsymbol{g}(\boldsymbol{W})$ and $\boldsymbol{Y'} = \boldsymbol{g}(\boldsymbol{W'})$ (see algorithm 1) where the columns of $\boldsymbol{Y}$ are given by $\boldsymbol{y}_j = \boldsymbol{g}(\boldsymbol{w}_j)$. Instead of applying the DMD with state evolution matrices as inputs, matrices $\boldsymbol{Y}$ and $\boldsymbol{Y'}$ i.e. containing evolution of the observables are used as inputs. In fact, DMD applied to the state trajectory/evolution is akin to using $\boldsymbol{g}(\boldsymbol{w}) = \boldsymbol{w}$

as the observable. A Theorem [220] connecting the DMD and Koopman analysis shows that the Koopman and DMD eigenvalues are identical if, 1) the set of observables is sufficiently large (higher chances of being in the span of Koopman eigenfunctions) and 2) the data is rich. In other words, choice of the observables plays arguably the most important role in its success.

Computing a finite-dimensional approximation of the eigenvalues and eigenvectors of the Koopman operator ($\boldsymbol{K}$) requires consideration of the fact that by definition a finite dimensional quantity will be an approximation of the infinite dimensional quantity $\mathcal{K}$. Careful choice of observables will dictate performance of the numerical procedure. Lastly, only the state evolution is known i.e. the system is a black-box. But since there is no principled method to select observables, some heuristics and/or expert knowledge must be employed. The numerical procedure essentially follows algorithm 1 applied in the space of observables. An important practical requirement for $\boldsymbol{g}$ is its ability to be inverted for recovery of the state ($\boldsymbol{w} = \boldsymbol{g}^{-1}(\boldsymbol{g}(\boldsymbol{w}))$). Figure 17 compares DMD with its Koopman generalization.

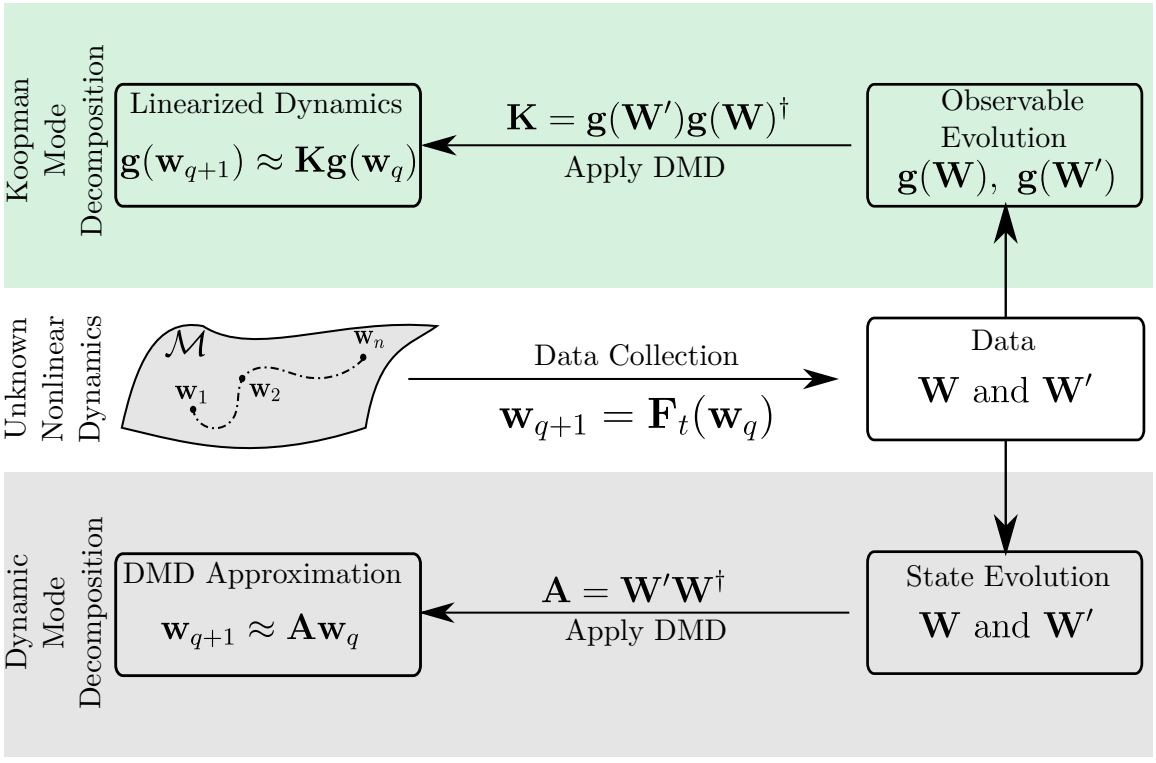


Figure 17: Koopman and Dynamic Mode Decomposition (adapted from [136])

Before proceeding, some desirable properties of DMD and its Koopman generalization are worth re-stating: 1) DMD provides an equation-free method to mimic the behavior of a system by finding a tangent linear approximation using data alone, 2) its Koopman generalization provides hope for linearizing a nonlinear system albeit contingent on the ability to identify appropriate observables, 3) DMD focuses on dynamics rather than L_2-norm (in POD), 4) DMD brings in an operator theoretic perspective to non-intrusive ROMs, and 5) the algorithm is endowed with a rank reduction step inherently using projection onto POD modes making it an ideal candidate for ROMs. As stated before, there has been no investigation into a parametric extension of DMD-based ROM despite its application in a wide variety of problems across disparate fields. However, an initial step leveraging the Koopman theory for creating a ROM for the parametric static Euler equation system has been recently demonstrated on an airfoil problem [213]. Although no elements of DMD are exploited, the method provides evidence through demonstration that Koopman theory based methods are worth pursuing for ROMs, especially the possibility of linearizing using nonlinear transformations of the state variables. The following section reviews the method in-depth.

2.4.4 Koopman-Based Method for Static Parametrized Systems

Developed by Renganathan et. al. [213] for static parametric systems, this method utilizes heuristics in an attempt to leverage the promise of *Koopman* theory to transform (or lift) and thereby linearize the discrete equation system in the observable space. It relies on knowledge of the governing equation in the continuous-PDE form. Beginning with identification of observables using the terms in the governing equation, an underdetermined set of transformed equations is formed followed by the construction of operators for the linear terms using an appropriate technique (finite difference, finite volume etc.). The underdetermined system of equations is closed using nonlinear analytical closed-form expressions that enforce consistency between the observables and the primitive variables. Projection is performed onto a basis set obtained separately for each observable. The method creates a database of ROMs in the *offline* step and interpolates between reduced order operators

on matrix manifolds (to preserve symmetric positive-definiteness) for unseen parameters using multivariate Lagrange interpolation in the *online* phase. Finally, solution of the ROM (in the reduced/latent space) is posed as an optimization problem that is solved using the sequential quadratic program (SQP).

Say a nonlinear static system has the following form,

$$\boldsymbol{f}(\boldsymbol{w}) = 0,\ \boldsymbol{f} : \mathbb{R}^m \to \mathbb{R}^m,\ \text{and}\ \boldsymbol{w} \in \mathbb{R}^m$$

The Koopman theory (see section 2.4.3) posits the existence of an infinite dimensional linear operator $\mathcal{L}$ (with its discrete counterpart $\boldsymbol{L}$) acting on some nonlinear function $\boldsymbol{g} : \mathbb{R}^m \to \mathbb{R}^{n_o m}$ mapping the state variables to observables. Assuming that this nonlinear map is known, we get

$$\boldsymbol{f}(\boldsymbol{w}) \to \boldsymbol{L}(\boldsymbol{g}(\boldsymbol{w})) \Leftrightarrow \boldsymbol{A}\boldsymbol{g}(\boldsymbol{w}) = \boldsymbol{f}_s$$

where $\boldsymbol{f}_s$ consists of the source and boundary condition terms. The authors in their work note that because nonlinear terms in the original system are transformed, the RHS does not depend on the state $\boldsymbol{w}$. Let $\boldsymbol{x} \leftarrow \boldsymbol{g}(\boldsymbol{w})$. The transformed linear system along with parametric dependence can be written as

$$\boldsymbol{A}(\boldsymbol{\mu})\boldsymbol{x} = \boldsymbol{f}_s(\boldsymbol{\mu}) \tag{38}$$

The ROM construction for the equation set recast in the observable space proceeds as follows. First, a heuristic that chooses the linear and nonlinear terms in the continuous PDE is used as a choice for the nonlinear state-to-observable map $\boldsymbol{g}$ or $\boldsymbol{x}$. Therefore, a system of p coupled system of PDEs containing p primitive variables and s linear and nonlinear terms gives rise to the following observable at parameter value $\boldsymbol{\mu}$,

$$\boldsymbol{x}(\boldsymbol{\mu}) = [\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_p^T, \boldsymbol{x}_{p+1}^T, \ldots, \boldsymbol{x}_{n_o=p+s}^T]^T|_{\boldsymbol{\mu}} \in \mathbb{R}^{n_o m}$$

where $\boldsymbol{x}_i \in \mathbb{R}^m$, and n_o is the total number of observables. A global POD basis set for each observable is then constructed separately using an *offline* evaluation of the full order model using an appropriate DoE. The following construction of the resulting POD basis ensures

that the projection is valid for eq. (38),

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}_1 & & \\ & \ddots & \\ & & \boldsymbol{\Phi}_{n_o} \end{bmatrix} \in \mathbb{R}^{n_o m \times K}$$

where $K = k_1 + k_2 + \cdots + k_{n_o}$ such that each observable may have its own truncation rank $k_j,\ j = 1, \ldots, n_o$. For the projection step, the method uses *Galerkin* kind on the normal equations (with the ansatz $\boldsymbol{x}(\boldsymbol{\mu}) = \boldsymbol{\Phi}\boldsymbol{x}_r(\boldsymbol{\mu})$) instead of the rectangular underdetermined system to get,

$$\boldsymbol{\Phi}^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{\Phi}\boldsymbol{x}_r = \boldsymbol{\Phi}^T\boldsymbol{A}^T\boldsymbol{f}_s$$

$$\text{or, } \boldsymbol{\Phi}^T\boldsymbol{B}\boldsymbol{\Phi}\boldsymbol{x}_r = \tilde{\boldsymbol{f}}_s$$

$$\text{or, } \boldsymbol{B}_r\boldsymbol{x}_r = \tilde{\boldsymbol{f}}_s$$

where $\boldsymbol{B}_r = \boldsymbol{\Phi}^T\boldsymbol{B}\boldsymbol{\Phi} \in \mathbb{R}^{K\times K}$ and $\tilde{\boldsymbol{f}}_s = \boldsymbol{\Phi}^T\boldsymbol{A}^T\boldsymbol{f}_s \in \mathbb{R}^K$. Note that dependence of the system matrices on $\boldsymbol{\mu}$ has been dropped just for convenience. In general, $\boldsymbol{B}_r$ and $\tilde{\boldsymbol{f}}_s$ may be dependent on the parameters. Finally, the following constrained optimization problem is solved to compute $\boldsymbol{x}_r$ using the SQP algorithm,

$$\underset{\boldsymbol{x}_r}{\text{minimize}}\ 0.5||\boldsymbol{B}_r(\boldsymbol{\mu})\boldsymbol{x}_r - \tilde{\boldsymbol{f}}_s(\boldsymbol{\mu})||_2^2$$

$$\text{subject to } \boldsymbol{h}(\boldsymbol{x}_r) = 0$$

where $\boldsymbol{h}(\boldsymbol{x}_r)$ is a nonlinear equality constraint that is problem dependent. Enforcement of consistency between the observables (which are functions of state) and the state variables is its main role. Note that the ability to prescribe appropriate equality constraints heavily relies on whether the analytical form of the governing equations are known. In addition, the SQP is solved for every parameter point in the DoE to yield a database of ROMs which is then exploited with interpolation to obtain reduced order matrices at unseen points in parameter space. The *offline* cost is dominated by $\mathcal{O}[n_o \times \min(m^2n, n^2m)]$ of the SVD required for computing the basis set. Solving the n SQPs also adds to the offline cost (which is hard to estimate but assumed to be negligible because it is independent of the full order dimension m). The *online* cost includes that of the interpolation to obtain the

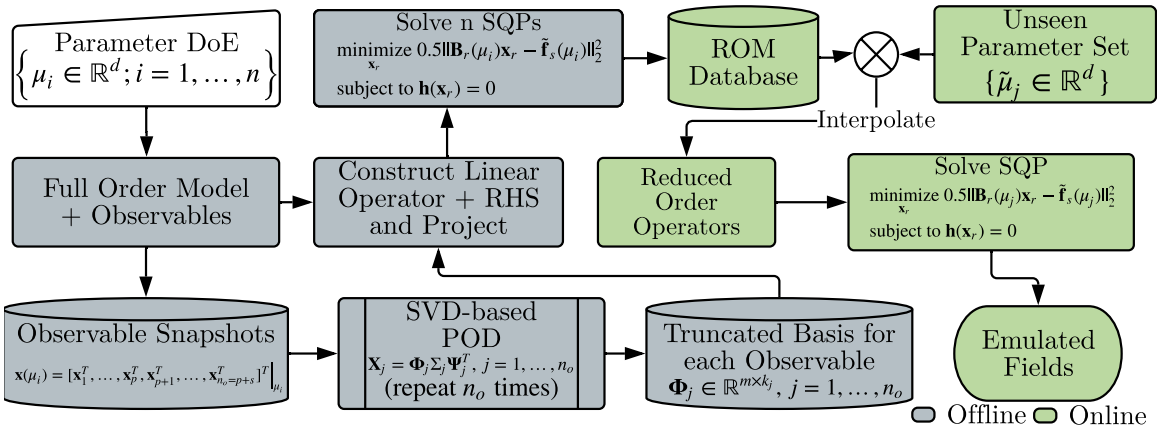


Figure 18: Offline-Online Decomposition in Koopman-Based Projection Framework

reduced order operators in addition to the solution cost of the SQP to solve the ROM. Figure 18 summarizes the method.

2.5 Strategies for Managing High-Dimensional Design/Decision Variable Spaces

As briefly discussed in chapter 1, high-dimensional design/decision variable or parameter spaces are ubiquitous in *many-query* applications. Several challenges need to be tackled especially when creating surrogates and/or ROMs with high-dimensional parametric dependence. Let us now briefly discuss classifications as described by Wang et. al. [231, 232] specifically for scalar, data-fit surrogate modeling in high-dimensions and draw observations to motivate our investigation.

Decomposition-based [145, 149] ideas partition the original problem into smaller, more manageable sub-problems. The resulting sub-problems are then handled separately. There are several nuanced methods that address the task that have developed into techniques over time. But an example suffices to make relevant observations. Multidisciplinary analysis and optimization (MDAO) [167] is a prime example of a decomposition-based approach. It leverages a natural product structure delineation to draw boundaries, albeit subjective, that define independent sub-disciplines/problems. It is obvious to see how the input variables automatically get partitioned by association to one or more small sub-problem(s). Tasks such as surrogate creation may now exploit the partitioning by considering each sub-problem as

an independent entity, each with relatively lower input space dimensionality. Decomposition-based methods for moderately complex systems rely on matrix decomposition-based/graph theoretic abstractions to yield favorable properties from the decomposed system. Although used successfully, the following issues exist: 1) subjectivity in the choice of boundary for the decomposition and 2) irreducibly large sub-problem input spaces (aero and structural subproblems may still have a large parameter space dimensionality).

Screening methods reduce dimensionality by exploiting sampled points to recognize and retain the most important inputs and their interactions while removing noise and other insignificant contributors to variability in the output (as indicated by the sampling). PCA and analysis of variance (ANOVA) [178] are common techniques to perform screening. Once the significant variables and interactions are identified, the other variables are simply dropped from consideration in the task (for instance, in the case of surrogate modeling). These methods are applicable to single and multiple outputs. While local sensitivity techniques indicate information akin to partial derivative, global sensitivity methods explain the variability of quantities over the entire domain of inputs. Generally speaking, screening methods are advantageous as a tool to remove noise and/or provide an insight into major input contributors to an observed output's variation. Consequently, the findings may be used to guide modeling efforts. For surrogate modeling, while they reduce dimensionality, there is a possibility of losing model accuracy by simply ignoring some variables (which may be actually present in the black-box truth model).

Mapping, as a term, has been used in a broad sense to include two kinds of techniques: 1) dimensionality reduction (see section 2.2) as a means to reduce input space and 2) mapping to create a link between small and large dimensional input spaces for optimization applications [25, 211, 215]. Our discussion is limited to the former since it is relevant and aligned to the theme in this work. These techniques use either linear or non-linear projection to transform a set of correlated input variables in high-dimension to a smaller, more manageable set. For reasons mentioned earlier (skipped for brevity, see section 2.2), linear methods such as PCA are better suited when compared to non-linear ones in practical applications. In the context of ROMs for inverse-design, authors [151] have applied PCA

to a sampled set of parameter points in order to reduce dimensionality. Following the reduction, the model is assumed to be restricted to the lower dimensional subspace. However, simply operating on samples of inputs does not consider the outputs' variation, i.e. the approach lacks the notion of goal. The PCA applied to gradients of scalar-valued outputs addresses this aspect. Originally conceived by Russi et. al. [223] and made popular in surrogate modeling and optimization communities by Constantine et. al. [72, 74], the so-called Active Subspaces extracts averaged directions of most variability in the input space by performing PCA on the gradient w.r.t. multivariate scalar functions. The method has seen successful applications in creating surrogates for high-dimensional scalar multivariate functions [35, 36, 159, 186]. Because it is relevant and related to the method developed in this work, let us briefly describe the theory behind the Active Subspace Method (ASM). For a detailed discussion, the reader is referred to [72].

The numerical procedure for exploiting the method for surrogate modeling begins with approximating the matrix $\boldsymbol{C}$ defined as

$$\boldsymbol{C} = \int (\nabla_{\boldsymbol{\mu}} f)(\nabla_{\boldsymbol{\mu}} f)^T \rho d\boldsymbol{\mu},\ \boldsymbol{C} \in \mathbb{R}^{d \times d}$$

where $f : \mathbb{R}^d \to \mathbb{R}$ is a multivariate scalar function of $\boldsymbol{\mu} \in \mathbb{R}^d$, $d \gg 1$, ∇ denotes the gradient, and ρ is the underlying probability density which governs $\boldsymbol{\mu}$. Since $\boldsymbol{C}$ is a continuous quantity, a practical recipe for number (n_∇) of gradient samples required for a good rank-k numerical approximation of the expected value of $\boldsymbol{C}$ is given by,

$$n_\nabla = \alpha k \log d$$

where α is an oversampling parameter (typically chosen between 2-10) and d is the number of components in $\boldsymbol{\mu}$. The idea central to the method is using the eigendecomposition of $\boldsymbol{C}$ to find directions along which on average the function f varies more than other directions with the hope that number k of such directions is small relative to d. The existence of an Active Subspace (AS) is determined by examining the spread among eigenvalues of $\boldsymbol{C}$. A gap between the k-th and $k+1$-th eigenvalue suggests that the Active Subspace is k-dimensional. Note that this is different from choosing the truncation rank for ROMs. The criterion for selecting the intrinsic dimensionality of the inputs depends on whether the

samples accurately represent the landscape of the function, and is somewhat imprecisely defined. However, it works well in practice. The eigenvectors that correspond to the first k eigenvalues define the Active Subspace. The scalar function is assumed to vary significantly only in this k-dimensional subspace. Therefore, instead of sampling in d dimensions one can sample in $k \ll d$ dimensions for creating a surrogate thereby effectively reducing the *curse of dimensionality*. Moreover, note that this is achieved by sampling n_∇ gradients which only has a logarithmic dependence on d. The procedure below may be followed for construction and exploitation of the Active Subspace:

1. Draw n_∇ i.i.d. samples $\{\boldsymbol{\mu}_i;\ i = 1, 2, \ldots, n_\nabla\}$ from the density ρ and evaluate gradient $\nabla_{\boldsymbol{\mu}} f(\boldsymbol{\mu}_i) \forall i = 1, 2, \ldots, n_\nabla$.
2. Form,
$$\tilde{\boldsymbol{C}} = \frac{1}{n_\nabla} \sum_{i=1}^{n_\nabla} (\nabla_{\boldsymbol{\mu}} f(\boldsymbol{\mu}_i))(\nabla_{\boldsymbol{\mu}} f(\boldsymbol{\mu}_i))^T \approx \boldsymbol{C}$$
3. Diagonalize $\tilde{\boldsymbol{C}} = \tilde{\boldsymbol{\Xi}} \tilde{\boldsymbol{\Lambda}} \tilde{\boldsymbol{\Xi}}$.
4. Inspect gap/jump eigenvalues in $\boldsymbol{\Lambda}$ to determine k and choose first k columns in $\tilde{\boldsymbol{\Xi}}$ (denoted at $\tilde{\boldsymbol{\Xi}}_1$).
5. Create surrogate model in the projected space $g(\boldsymbol{\mu}_k) = f(\tilde{\boldsymbol{\Xi}}_1 \boldsymbol{\mu}_k),\ \boldsymbol{\mu}_k \in \mathbb{R}^k$, or viewed as a proxy g such that $g(\tilde{\boldsymbol{\Xi}}_1^T \boldsymbol{\mu}) = f(\tilde{\boldsymbol{\Xi}}_1 \tilde{\boldsymbol{\Xi}}_1^T \boldsymbol{\mu}) \approx f(\boldsymbol{\mu})$.

Several benefits are offered by this method when compared to other methods mentioned above: 1) its reliance on gradients to find objective oriented directions, 2) logarithmic dependence on input space dimensionality for number of gradient samples required, and 3) the property that no design variable needs to be dropped (like in screening), instead only some directions in the high-dimensional input space are not considered. Note however that when concerned with multiple outputs as in our case, it is unclear as to how the described method can be used to reduce dimensionality in the input space. Moreover, the requirement for availability of gradients of the field outputs w.r.t. to the inputs is harsh and computationally demanding. Nevertheless, the method is a suitable candidate because of its desirable properties when gradients are available.

2.6 Randomized Linear Algebra to Handle Large Matrices

An often overlooked but important step in the generation of parametric reduced order models is the dimension reduction/basis computation step. As mentioned before the workhorse for most non-intrusive ROM methods is the POD whose computational complexity measured in terms of flop counts scales as $\mathcal{O}(\min(m^2n,\ mn^2))$ for a matrix having row and column dimensions m and n, respectively. A common argument for disregarding the POD step is the relatively high computational cost of evaluating the full order model when compared to that of the POD step. However, in cases where the number of parameters/design variables is large and/or the full order model is time-dependent, computation of the POD can become intractable due to the resulting large snapshot matrices. Storing and accessing the large matrix in-memory also become cumbersome and sometimes impossible on personal computers. All the aforementioned problems manifest as a paralyzing bottleneck, especially when data is generated using black-box tools that are parallelized for optimal performance on a cluster to create a parametric ROM on a personal computer.

Dimensionality reduction through POD is an unavoidable, expensive step in the *offline* phase of generating parametric ROMs for large problems. Solutions to alleviate this computational burden to enable practical applications must be aimed at reducing both the computational complexity and storage requirements while still performing the dimension reduction step in a reasonable amount of time.

As mentioned before, one potential alternative strategy to handle large matrices that do not fit in-memory is to utilize the method of snapshots where the size of the matrix of snapshots $\boldsymbol{W} \in \mathbb{R}^{m\times n}$ is reduced to $\boldsymbol{W}^T\boldsymbol{W} \in \mathbb{R}^{n\times n}$. However, performing the matrix multiplication scales with the worst case time complexity of $\mathcal{O}(n^{2.3737})$ even for the best algorithm today. For very large matrices, in addition to the comparable cost of the multiplication step with the SVD, the matrix-matrix product is prone to introduce round-off errors in $\boldsymbol{W}^T\boldsymbol{W}$. However, the round-off error is likely to be relatively inconsequential in comparison to the cost of the matrix multiplication for matrix sizes typically found in large scale problems.

A potential appealing candidate for addressing the issues discussed above is the class

of so-called randomized matrix decomposition techniques recently developed primarily by Halko et. al. [112]. The approaches rely on random projections to efficiently build a subspace that spans the range of the input matrix resulting in a method that provides a faster computation time while sacrificing some accuracy. Succinctly put, randomized algorithms require $\mathcal{O}(mnk)$ (or in some cases $\mathcal{O}(mn \log k)$) flops for a rank-k reduction albeit at a loss of minimal accuracy in the low-rank approximation. Alternative strategies to compute partial factorizations based on incremental SVD, column pivoted QR or Arnoldi with comparable costs exist, but these algorithms can be challenging to implement on modern memory and communication-constrained hardware [22, 193].

Another line of work closely related to compressed sensing is referred to as matrix sketching based algorithms [89]. These methods borrow from the theory of compressed sensing and rely on sampling specific rows of the input matrix to construct a low-rank approximation. As one can imagine, these methods can be extremely useful in memory constrained situations. They are similar to randomized methods in that the low-rank approximation suffers some loss in accuracy. However, the rich theory of compressed sensing provides good certificates on the quality of the approximation.

It is natural to ponder whether they are applicable in the context of ROMs in general. The past couple of years have witnessed a slow adoption of randomized techniques for performing dimensionality reduction. Several researchers [10, 41, 42, 49, 90] have successfully used the randomized SVD (rSVD) [168] to compute the POD projection subspace for use with non-parametric ROMs. Very recently, researchers [22] have compared performance of various randomized algorithms for compression of data from numerical simulations. Sketching-based approaches have not even seen use for non-parametric ROMs despite their desirable traits. First promising efforts for data compression using sketching are underway [252]. Neither of the approaches discussed above has surprisingly received no attention for the creation of parametric ROMs. We assert that, if successfully applicable, then randomized techniques (either sketching-based and/or random projection-based) will enable parametric ROM creation for truly practical problems, mainly because these algorithms are designed to compute efficient factorizations while minimizing the loss in accuracy introduced

due to randomization.

In order to give a feel for how randomized factorization works, the steps involved in rSVD for a rank-k reduction of matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ are mentioned below. Specific use with parametric ROMs will be introduced and detailed in chapter 4. The idea for constructing low-rank approximations using randomized schemes can be split into two steps: 1) build a low-dimensional subspace to capture the operation of the input matrix and 2) constrain the matrix (i.e. decrease the size) to the subspace found in the previous step and factorize it appropriately using standard deterministic algorithm. The following lists an intuitive stencil to compute a rank-k approximation using the two steps.

1. Draw a random Gaussian matrix $\boldsymbol{G} \in \mathbb{R}^{n \times (k+p)}$. Here p is called the oversampling parameter.
2. Form $Y = \boldsymbol{AG} \in \mathbb{R}^{m \times (k+p)}$. $\boldsymbol{Y}$ should ideally represent the range of $\boldsymbol{A}$ and is thus called the sampling matrix.
3. Find the column-space $\boldsymbol{Q} \in \mathbb{R}^{m \times (k+p)}$ of $\boldsymbol{Y}$ using $\boldsymbol{Y} = \boldsymbol{QR}$.
4. Create a small matrix $\boldsymbol{B} = \boldsymbol{Q}^T A \in \mathbb{R}^{(k+p) \times n}$ i.e. projection of $\boldsymbol{A}$ onto Range($\boldsymbol{Y}$).
5. Compute (inexpensive) SVD of $\boldsymbol{B} = \tilde{\boldsymbol{U}} \boldsymbol{D} \boldsymbol{V}^T$.
6. Recover approximate left singular vectors $\boldsymbol{U}$ as $\boldsymbol{U} = \boldsymbol{Q}\tilde{\boldsymbol{U}}$.

Note that $\boldsymbol{A} \approx \boldsymbol{UDV}^T$ is akin to an approximate SVD. Although the computational complexity of the method is $\mathcal{O}(mnk)$ (identical to several other partial algorithms), reliance of the algorithm on random matrix-matrix multiplications offers desirable advantages such as a small practical execution time (high speed) and ease of use in situations where the data is stored on a hard drive or distributed across various computers. In fact, the complexity can potentially be further reduced to $\mathcal{O}(mn \log k)$ by designing structured sparse random sampling matrices for $\boldsymbol{G}$ instead of using a Gaussian matrix [252].

2.7 Summary of Observations and Research Objective Formulation

The topics discussed above were meant to further scope down and establish the main research objective to answer the motivating question posed in chapter 1 which is restated below.

Motivating Question

How can *non-intrusive parametric reduced order modeling methods* be improved to enable the creation of surrogates to approximate *high-dimensional fields* emanating from *PDE-based black-boxes* to enable *many-query* exercises encountered in *practical* engineering design and analysis problems?

Note that the arguments justifying the focus on non-intrusive parametric ROM methods were stated in chapter 1. The term *practical* has been used to signify problems that have high-dimensional field outputs and are characterized by high-dimensional design/decision variable spaces often leading to large snapshot matrices. In this chapter, the state-of-the-art in non-intrusive reduced order modeling methods (some parametric, others non-parametric) was reviewed to identify the shortcomings that have prevented their successful adoption for practical engineering problems. Approaches to tackle high-dimensional parameter spaces, specifically with non-intrusive ROMs were largely found to be lacking as corroborated by several researchers. Therefore, new techniques or improvements to existing approaches are required to advance the state-of-the-art and facilitate their adoption for applications involving practical engineering problems. Given these high-level observations, the main research objective tackling challenging aspects that have received limited attention in literature so far can be re-stated as follows.

Research Objective

Develop and/or enhance *non-intrusive parametric reduced order modeling methods* to address challenges that limit their viability in many-query contexts involving practical engineering design problems with a focus on analyses that:

1. are expensive to evaluate,
2. output high-dimensional field quantities, and
3. are characterized by a relatively large number of design variables.

In order to further clarify the stated research objective and formulate specific research questions, the assumptions and/or requirements that define various elements of the statement are stated:

1. **Regarding Black-Box PDE-based Systems:** By definition, non-intrusive parametric reduced order models are applicable to black-box systems whose outputs are high-dimensional fields from PDE-based governing equations. For this work, it is assumed that the form of the PDE equation is unknown.

2. **Regarding Large State/Field Vector Output:** Usage of the term "practical" while referring to *many-query contexts* in engineering design implies that the high-dimensional fields must be in the order of $\mathcal{O}(10^{5-6})$. Even though it may seem like a trivial requirement to impose explicitly at first sight, even a moderate number of dimensions can result in impractical dimension reduction related *offline* costs for parameter spaces of moderate size.

3. **Regarding Large Design/Decision Variable Spaces:** As mentioned previously, the few studies that deal with large design variable spaces are *intrusive* parametric ROM methods. In this work, we are concerned with moderate to large ($\mathcal{O}(10100)$) parameter spaces. This assumption is important because in most of the realistic modern day engineering problems, detailed parametrizations provide a mechanism to ensure thorough exploration of the design space to push the envelope of achievable objectives thus making them unavoidable.

Before formulating the specific research questions to achieve the main objective, what follows is a summary of observations made throughout this chapter. A natural segmentation induced by the research objective is used to categorize these observations into ones related to 1) non-intrusive ROM methods and challenges associated with large *offline* costs due to dimension reduction and high analysis execution times and 2) treatment of high-dimensional parameter spaces. The observations will be made such that they directly identify gaps, justify choices and help in proposing novel areas of inquiry.

Non-Intrusive ROM Methods: Observations

- POD-based methods have seen immense success in literature for steady and unsteady parametric systems. However, as noted earlier, their only handicap is the inability to capture discontinuities. Despite several applications in the literature (see Table 2), the method has been mostly applied on canonical and engineering problems with a handful of parameters.

- Data-driven operator inference, a promising method, has been demonstrated to work well only on problems with $d \ll 10$ parameters. Since the inference problem cost scales exponentially with the degree of nonlinearity, the method is computationally feasible only for systems with low order polynomial terms and requires knowledge of the discretized form of the governing equation.

- While POD-based methods focus on minimizing the reconstruction error in the L_2-norm (or energy), the DMD, a purely data-driven method, focuses on the dynamics instead while adopting the operator theoretic perspective to derive a linear tangent approximation to the system. DMD is completely equation-free and does not require knowledge of the governing equations or any a-priori assumptions.

- Although DMD is unreliable for nonlinear systems, leveraging the Koopman theory for linearization allows its usage with governing equations consisting of arbitrary nonlinearities. Researches have proposed heuristics [138, 179, 213] and machine learning [135, 148, 263] to show success of the DMD procedure for nonlinear systems.

- Despite the immense hope and desirable properties of DMD and its Koopman connection, its usage for parametric ROMs has received no attention as reported by several researchers [15, 191].

Non-Intrusive ROM Methods: Observations

- A first step in the reliance on Koopman theory for linearization in the parametric ROM context was demonstrated by Renganathan et. al. [213]. However, the study did not depend on the DMD and was limited to static systems. Construction of the linear operators relied on knowledge of the governing equation. This study was a pivotal step in developing a parametric ROM using some elements of Koopman theory albeit for static systems.

- *Offline* costs, excluding evaluation of the full order model, is an often neglected bottleneck in terms of data storage, read/write operations and numerical computation of the POD when considering large, time-varying fields dependent on a large number of parameters. This hinders easy application and implementation on truly large problems.

- Recently developed, randomized techniques relying on sketching and random projections can afford computation of the SVD at a significantly lower computational complexity and smaller memory footprint in exchange for some accuracy. Albeit still nascent, their use for non-parametric ROMs has started to gain traction. Their adoption for use with parametric ROMs has received limited attention in the literature so far.

The few final observations regarding *offline* costs are relevant in the process pipeline for constructing ROMs. They shed light on major practical challenges that prevent realization of the full potential of ROMs. Therefore, this work asserts that it is imperative to address them.

A couple of important issues concerning tackling parametric dependence in ROMs (as discussed in section 2.3) that lack attention in the literature as reported by [31] and otherwise noted in our discussion are now stated as observations:

Parametric Dependence: Observations

- The literature lacks attention to systems with high-dimensional parameter spaces where $d \geq 10$ (see Table 2). Intrusive parametric ROMs with moderate parameter dimensions have received some attention [51, 151]. However, we are unaware of studies on problems with such characteristics in the realm of non-intrusive parametric ROMs.

- While several techniques for handling surrogates with large parameter space dimensions exist, the Active Subspace method seems to have arguably been the most impactful and successful for scalar-valued functions in practice. Its usage for vector-valued outputs has started receiving attention recently. Of particular merit is its property that the number of samples required is proportional to the logarithm of the number of input parameters. However, the requirement to know gradients is rather restrictive, crippling even, in the case of prediction of field quantities.

- A single best method to interpolate between ROMs does not exist. Different approaches are applicable based on characteristics of the problem at hand. There clearly exists a lack of comprehensiveness in identifying performance of different interpolation methods for both intrusive and non-intrusive ROMs especially when the number of parameters is large, mostly because very few studies can be found in literature.

The observations clearly indicate that there is a need to evaluate the applicability of existing state-of-the-art non-intrusive parametric ROM methods from the perspective of big datasets and large parameter spaces. The next chapter will formally pose some questions while identifying the candidate ROM method on which the developments in this work will be demonstrated. Finally, the identified challenges will be used to segregate the work into research areas, in each of which we will pose questions to help narrow the scope, delve further into the literature, and formulate and present our hopefully novel contribution.

CHAPTER III

RESEARCH FORMULATION

Let us begin by briefly discussing the stencil that this works adheres to for guiding the process of formulating the research pursued in this work.

1. First, using the scoped overarching objective and observations/gaps made in the literature review, broad challenge areas will be identified. The justification for pursuing these avenues will largely depend on either the need to improve a method by adding a capability or address a deficiency based on the background in chapter 2.

2. Second, specific research questions to answer identified issues will be posed, if necessary, for specific aspects.

3. Finally, a set of arguments justifying why each research question is important will be built alluding to a hypothesis, wherever possible, without explicitly stating it or going in-depth in this chapter. The subsequent chapters will actually present the experiments performed in this work in detail.

Remark: Some arguments and statements presented in this chapter may seem inopportune. Rest assured that in the following chapters, each question will be presented again with the pertinent additional literature search, appropriate arguments for formulating a hypothesis, the hypothesis itself, and accompanying experiment(s).

3.1 Main Research Objective

Given the observations noted through the literature review to identify challenges and opportunities, the overarching research objective is restated here.

Research Objective

Develop and/or enhance *non-intrusive parametric reduced order modeling methods* to address challenges that limit their viability in many-query contexts involving practical engineering design problems with a focus on analyses that:

1. are expensive to evaluate,
2. output high-dimensional field quantities, and
3. are characterized by a relatively large number of design variables.

The rationale for this research objective is driven by a lack of attention to the requirements warranted by modern day applications, as identified in the literature. One, legacy PDE-based black-box codes are used in most practical situations, and therefore we have no option but to pursue non-intrusive methods that exclusively rely on data. While methods [191, 213] that assume and exploit knowledge of the terms in the governing equations exist, they fall short when the underlying analysis is a black-box in the true sense of the term, i.e. there is no knowledge of the underlying equations that produce the field outputs. Two, realistic problems are ones with a naturally occurring high-dimensional design parametrization. As shown in table 2, most applications in the literature deal with only a couple of input parameters. Three, ROMs for analyses with large field outputs are challenging to compute as evidenced by a sparse number of studies that tackle large fields in the literature (see table 2). Four, high-dimensional outputs usually imply a large analysis execution time. In such cases, the conflict between model predictive accuracy and high-offline cost attributed to large execution time becomes an obstacle for constructing surrogate models, especially ROMs. Therefore, the presented observations can be partitioned into the following specific areas: 1) large field outputs, 2) large execution times and large field outputs, and 3) high-dimensional parameter spaces.

3.2 Research Area 1: High-Offline Cost for Dimension Reduction

One of the neglected problems we identified in the literature search was related to the offline computational cost associated with the dimension reduction step in the construction of parametric ROMs. The current state-of-the-art primarily relies on using the method of snapshots requiring a large matrix multiplication for which even the best algorithm today has an approximately squared polynomial complexity; which is comparable to the time complexity of the deterministic SVD. The need to investigate, identify and tackle specific issues regarding the offline cost due to the dimension reduction step motivates this first challenge area. We are now ready to state the overarching question this research area aims at investigating as follows:

Research Question 1

How can the *offline* cost, as measured by time complexity associated with the dimension reduction step, be decreased to enable efficient and sufficiently accurate construction of parametric predictive ROMs?

A source for potential confusion must be cleared before proceeding. Data generation and dimension reduction both contribute to the offline cost in the context of creating parametric ROMs. In fact, especially in the case of ROMs, one may argue correctly that when considering computationally expensive analyses, generating data is typically significantly more expensive than performing the dimension reduction step. Although there is truth in the claim, it may be argued that handling large datasets on personal and mobile computers for truly large systems is an important emerging need for enabling digital twin applications where real-time construction of ROMs [124, 163], by processing large data streams from sensors, is of interest. Therefore, developing the ability to decrease dimensionality of large datasets at least relatively efficiently, if not on-the-fly, is of utmost importance especially for predictive parametric ROMs.

Goal. The primary goal of the work under this research area is the evaluation of the viability of randomized variants of the SVD algorithm to perform the dimension reduction step and efficiently compute approximate but sufficiently accurate subspaces for constructing parametric ROMs. We will see later that randomization requires specification of certain parameters that control the trade-off between incurred loss in accuracy and gains in the computational efficiency. In our pursuit, focus will be given on developing a strategy that requires minimal specification of additional algorithm specific parameters. Recently proposed pertinent SVD algorithms will first be chosen as candidates for dimension reduction. Because the final goal of any parametric ROM is the fast prediction of a field quantity, the performance of the randomized algorithms as a means to enable ROM construction must be evaluated against the predictive capability of a ROM built using the deterministic SVD-based POD computation algorithm.

As for the choice of the specific ROM method on which the randomization strategies will be tested, any of the non-intrusive ROM methods detailed in chapter 2 may technically be chosen as a test bed for techniques developed under this research area. We choose the POD and interpolation-based method because, they have displayed immense success in practice. It was also noted that the immense success of non-intrusive POD-based parametric ROMs may be owed to POD's several advantages including but not limited to its ability to handle time-dependence, its reconstruction error minimizing property in the L_2 sense, its completely equation agnostic formulation, and the predictive capability of the resulting ROMs. Because of the equation-agnostic trait, the POD and interpolation-based methods enjoy the advantage that they enable the creation of models for only a subset of state variables because the coupling between variables need not be considered. Moreover, one potential concern with choosing other candidate nonintrusive ROM methods is that most of them have challenging requirements such as interpolation of matrices, basis vectors, etc. For instance, even though the operator inference- and DMD (Koopman)-based approaches may seem like promising candidates, they involve interpolating amongst matrix quantities which is a challenging endeavor to begin with. Such challenges have the potential of confounding the effect of randomization on the performance of the ROM, which must ideally be isolated

for clear insight. In contrast, with the POD and interpolation approach, the steps following dimension reduction are simple and straightforward in the sense that there are no stringent requirements on the regression/supervised learning step.

It must be noted with caution that the POD is known to struggle with fields that are characterized by complex nonlinear features such as shocks, discontinuities etc. However, since the source of the deficiencies is the POD step itself, it is reasonable to expect that the deficiency will manifest identically in any comparison between methods using the variants of the POD. In fact, identically manifesting deficiencies can also serve as corroboration that the randomized variants are performing as expected.

Based on the facts regarding the use of randomization as a strategy for the dimension reduction step in chapter 2, we are now ready to state the hypothesis for the main research question as follows:

Hypothesis 1: *Techniques from Randomized Linear Algebra (RandNLA) will effectively decrease the computational cost associated with the dimension reduction step while incurring a reasonable penalty in terms of accuracy, thereby enabling efficient construction of parametric ROMs for large systems.*

The validation of the main hypothesis can be further partitioned based on the steps involved in the construction of POD and interpolation-based ROMs. The first ingredient is the computation of the POD subspace i.e. performing dimension reduction. Although whether a basis set is accurate enough for constructing a ROM will eventually be indicated by its predictive capabilities, before directly building a model, the quantitative discrepancy between the true subspace and approximate subspace when randomization is used must be analyzed. Furthermore, the relationship between the algorithm specific parameters and their effect on the discrepancy will also provide insight into their relationship with (or sensitivity to) the characteristics of the dataset. The evaluation and quantification of this discrepancy is the purpose of the following research question:

Research Question 1.1

Can randomized variants of the SVD yield basis vectors that are suitable for construction of parametric ROMs?

Chapter 4 first begins with an in-depth introduction to randomized linear algebra techniques for performing dimension reduction and presents some candidate algorithms to perform tests on. We will identify and choose two recent algorithms that enable efficient calculation of the POD basis. Through tests on a canonical problem, the experiment will measure 1) the discrepancy between the basis vectors obtained from the deterministic and randomized algorithms and 2) the reconstruction error as the randomized algorithm specific parameters are changed. The randomized algorithms tested in this experiment will require specification of the truncation rank a priori rather than the *Relative Information Content* (RIC). Since the goal here is to purely compare and analyze the effect of randomization on the basis vectors, the requirement for specification of rank is desirable since it helps us fix the amount of data compression that must be performed by the randomized algorithm, allowing a one-on-one comparison. The outcome of this experiment will indicate how the the discrepancy between basis vectors changes with algorithm parameters and whether it is sufficiently small. Consequently, it will equip us with an intuition for setting the algorithm specific parameters depending on the physical traits of the problem-at-hand should we choose to leverage randomization in the construction of ROMs.

Note that the outcome that discrepancy is sufficiently small is not an objective one. Therefore, in order to objectively ascertain the viability of ROMs constructed for a fixed truncation rank (specified a priori) using randomized algorithms, we ask the following question:

Research Question 1.2

Are parametric ROMs constructed using randomized algorithms for the SVD step competitive in terms of predictive accuracy when compared to ROMs constructed using the deterministic SVD algorithm?

The primary purpose for this question is to compare the predictive accuracy of ROMs resulting from deterministic and randomized SVD-based POD computation. The term "competitive" is used to account for the deficiencies of the POD and interpolation-based ROMs that may be present in the case of analyses that have complex discontinuous flow fields. In other words, if the experiments for this question show that resulting ROMs have similar looking predictive error metric profiles on average, randomized algorithms are successful in serving as efficient basis computation techniques.

Up until this point, recollect that a stringent requirement for the candidate randomized algorithms has been the specification of the truncation rank a priori. When deterministic SVD is used, the truncation rank is computed using the RIC (an intuitive input parameter that indicates the amount of variance captured by the factorization). While subject matter experts can loosely classify problems as high/low rank, it is unreasonable to expect specification of the truncation rank a priori because it is impossible to know the exact truncation rank for a good reconstruction error. Therefore, to incentivize the adoption of randomized algorithms by specifying the RIC instead of the truncation rank, we pose the following final question for this research area:

Research Question 1.3

Can an approximate POD basis be efficiently computed by specifying the RIC instead of the truncation rank? If so, what is its effect on the computational savings offered by randomization? Does this strategy produce competitive predictive ROMs?

The pursuit of this research question will result in an investigation of the so-called *fixed-precision* randomized matrix factorization schemes. Given an input matrix, instead of specifying a truncation rank, these algorithms compute a factorization such that a prescribed error tolerance is met. If successful, the resulting randomized dimension reduction algorithm should ideally allow for the specification of the RIC and compute the truncation rank automatically. This question will 1) assess the effectiveness of *fixed-precision* algorithms in terms of basis discrepancy metrics, 2) quantify whether relaxing the specification of the

truncation rank unreasonably diminishes the computational benefits of randomization, and 3) evaluate how the resulting ROMs compare with their deterministic counterparts in terms of predictive accuracy.

By resolving the three questions posed above, we hope to validate the main hypothesis for this research area and propose a new method that can efficiently construct competitive ROMs for large problems under constrained computational resources by leveraging randomization in the dimension reduction step. All the experiments will first be performed on a representative canonical problem. The benefits offered by randomization will be finally highlighted by repeating the experiments on two practical flow problems.

3.3 Research Area 2: High-Dimensional Inputs

The second focus area is motivated by the challenges associated with handling a large number of input parameters when constructing surrogate models, especially ROMs. Evidence in the literature for POD and interpolation-based ROMs (see table 2) clearly shows that most applications, both academic and practical in nature, only focus on creating models with a couple of input parameters. Several researchers have also explicitly pointed out the need for surrogate modeling methods that can accommodate a large number of inputs [154, 231, 232, 255], especially for parametric ROMs [31]. Although our efforts are motivated by the need to enable ROMs with high-dimensional inputs, we will see later that the techniques developed in this dissertation are applicable to scalar surrogate models as well. As discussed in chapter 2, the taxonomy of strategies to tackle surrogates in high dimensions can be partitioned into decomposition-, screening-, and mapping-based techniques. Of most relevance to this research are techniques that fall under the mapping category. We briefly discussed some successful attempts such as the application of the PCA on sampled inputs, the Active Subspace method (ASM) etc. and pointed out their advantages and disadvantages. However, since ROMs by definition deal with emulating spatially distributed vector-valued quantities, it is not trivial to directly extend the described techniques to reduce dimensionality in the input space. We pose the following overarching question for this research area:

Research Question 2

How can the challenges posed by high-dimensional input spaces be tackled to enable construction of parametric ROMs with a relatively large number of inputs?

The use of the phrase "relatively large number of inputs" must be clarified before proceeding. Because applications involving parametric ROMs with a large number of inputs are sparse, our preliminary efforts will first focus on developing a method that successfully works with problems that have a relatively larger number of inputs when compared with the currently existing applications.

Goal. The primary goal of the work under this research area is the development of methods to construct nonintrusive ROMs for problems characterized by a large number of inputs given a pre-evaluated set of input-output pairs. We will see later that currently existing methods have paid limited attention to several issues related to sample sizes, unavailability of gradients, unstructured samples, etc. In what follows, we briefly present a summary of these challenges and mention how the current state-of-the-art fails to deal with them. Key issues regarding these challenges will be used to sharpen the scope further. In the work done under this research, our pursuit to develop a strategy that accommodates all the additional requirements/constraints will lead to a new kind of surrogate model that simultaneously regresses the relationship between input-output pairs and finds a low-dimensional input subspace.

The choice of the particular ROM method on which the developments in this research area will be tested is the POD and interpolation method. The rationale for this choice is largely in-line with the arguments made for its choice in the previous research area. As one can imagine, when dealing with any kind of parametric ROM, an attempt to accommodate numerous inputs (or any challenge related to input parameters for that matter) eventually reduces to the interpolation/regression step that is directly responsible for encoding parametric dependence. A ROM method that involves steps such as interpolation of matrix-valued quantities (as in operator inference-based methods) only add to the already

challenging task of handling a large number of inputs. One may argue that element-wise interpolation of the entries in the matrix can simplify the problem and make operator-based methods tempting candidates. However, we argue that one cannot control the size of the matrix to be interpolated. Even a projection subspace of modest size can lead to a situation where a large number of entries in the reduced operators must be interpolated as a function of a large number of inputs. With the POD and interpolation-based method in contrast, there are no specific requirements for the supervised learning step other than ensuring good predictive accuracy. Therefore, it is our candidate of choice.

Formulation of the hypothesis for the overarching research question follows from the discussion on the choice of the mapping-based techniques for tackling large input spaces in surrogate modeling tasks. Recall that within the mapping-based schemes, techniques based on dimension reduction in the input space were found to be most suitable and successful in practice. Therefore, it stands to reason that ideas within the contemporary methods can be improved, modified, and tailored for constructing ROMs by specifically focusing on addressing the constraints/requirements posed by vector-valued spatially distributed datasets. The hypothesis for this research area is stated as follows:

Hypothesis 2: *Parametric ROMs for a relatively large number of inputs can be constructed by modifying and leveraging approaches that identify and construct surrogates in a low-dimensional input subspace with a special focus on addressing the specific challenges posed by vector-valued field outputs.*

The answer to this research question will be developed in detail in chapter 5. In order to provide structure to the problem formulation and facilitate presentation of the research questions posed ahead, we briefly discuss the requirements for handling large inputs from the perspective of creating ROMs exclusively using input-output pairs.

First, as mentioned before, the recently proposed Active Subspace Method (ASM) requires the gradients to find directions in the input space in which the function varies more on average than others. If such a subspace exists and is low-dimensional, it can be used to reduce the input space dimensionality. When dealing with parametric nonintrusive ROMs

however, the quantities being interpolated are either projected coordinates in the POD subspace or projected inferred operators. The ASM would require the gradients of either the projected coordinates or each of the entries in the reduced operators with respect to each of the inputs. Obtaining these gradients entails intrusive modification of the source code. This dissertation assumes data is only available in the form of input-output pairs. Therefore, we require a method that does not rely on gradients. As such, the ASM is inapplicable.

Second, as we will see in chapter 5, all the *gradient-free* methods that currently exist for handling large input spaces heavily rely on a structured sampling plan. Simply put, their ability to discover a low-dimensional input subspace heavily depends on the choice of the points queried in the input-space. In fact, the methods dictate the sampling requirements to get a good estimate of the subspace, assuming it exists. Therefore, they are simply inapt when presented with a pre-populated set of inputs and outputs. We argue that it is common in engineering design to be confined to work with a set of designs and therefore we must develop a method that handles a large number of inputs and works efficiently with a pre-evaluated set of designs.

Third, a detailed literature search will also reveal that the gradient-free methods that mandate a structured sampling usually require a large number of expensive function evaluations to get a good estimate of the low-dimensional input subspace. This is primarily because most of the methods try to approximate the gradient of a high-dimensional function and apply the ASM. However, since ROMs are typically built for computationally expensive analyses, the requirement for a large number of samples is simply unrealistic and prohibitive. Therefore, it would be beneficial if a method is at least as competitive as the ASM in terms of the number of samples required to accurately find a low-dimensional input subspace.

In an attempt to develop a solution that satisfies all the requirements discussed above, this dissertation formulates a novel machine learning model that simultaneously finds a low-dimensional input subspace and trains a Gaussian process regression (GPR) model in it. The low-dimensional input subspace is parametrized using the Grassmann manifold, which is a manifold of linear subspaces. The optimization of the cost function is posed as a minimization problem on a product manifold that treats the hyperparameters of the GPR

model as members of the Euclidean space and the low-dimensional subspace as a member of the Grassmann manifold. Chapter 5 discusses the relevant literature at length, formulates, and presents the details of the proposed machine learning model.

Going forward, all the research questions will be aimed at evaluating the performance characteristics of the new manifold optimization-based machine learning model, identifying another state-of-the-art method and comparing its performance with our model's performance, and finally proposing a new method to construct POD and interpolation-based ROMs using the new machine learning model. Since this research area is different from the previous in that a new method is proposed, the research plan must first thoroughly evaluate whether the model successfully performs the basic tasks it is designed to accomplish. Next, a detailed analysis of its performance must be carried out. Once the traits of the method are clearly understood, we may confidently proceed with applying it to the construction of ROMs.

Mapping-based methods for surrogates in high-dimensional spaces have two primary goals:

1. the identification of a low-dimensional subspace that captures the majority of the function's variation (and its size, if possible), assuming it exists, and
2. training a surrogate model with good predictive accuracy in the low-dimensional subspace.

The purpose of the first research question is to ascertain and verify if the proposed machine learning model is able to recover a meaningful low-dimensional input subspace. Formally stated, it reads as follows:

Research Question 2.1

Can the proposed method successfully recover a meaningful low-dimensional input subspace, i.e., one that captures a majority of the true function's variation?

Once the method is developed, this question will be tested through an experiment that increases the dimensionality of the low-dimensional subspace progressively and records

whether the model improves its predictive accuracy as it is given additional dimensions. The experiment assumes that a low-dimensional subspace exists, fixes the values of its size, slowly increases it, and trains models for each size. Because there is no formal definition for the subspace the proposed model finds, as in the case of the ASM, we are bound to rely on the predictive performance of the model as an indicator of the effectiveness with which the subspace captures the variation of the function. The idea behind the experiment is as follows: first evaluate whether the model prefers meaningful directions rather than random directions given a size, then check whether the model's predictive accuracy increases as the number of dimensions increases. In chapter 5, we will build up arguments to justify this experiment and build a hypothesis that uses the minimization of the cost function to make a statement about the expected outcome. If successful, this experiment will validate the hypothesis and establish that the method produces a good subspace given the size of the lower-dimensional input subspace.

Even though the previous question establishes that the subspace found by the proposed model is sensible, we must try to investigate and quantify its relationship to some known, optimal subspace. The model's performance must be compared to other existing techniques that achieve the same goal under identical requirements/constraints. Finally, the model's performance on problems with different input dimensions, sample sizes etc. must be evaluated thoroughly. The following question attempts to quantify these relationships:

Research Question 2.2

Does the subspace recovered by the proposed model have any relationship to the Active Subspace? How competitive is the model's capacity to identify the subspace and yield good generalizability with respect to:

1. the number of inputs
2. the number of training samples
3. the size of the actual low-dimensional input subspace?

Note that the question presumes that the best subspace to compare with is the Active Subspace. In the mapping-based schemes, the only other techniques that computes a low-dimensional input subspace is the method that applies PCA to inputs sampled from the design space. The relationship of the outputs to the inputs is totally ignored. The ASM on the other hand exploits the relationship using the gradients and performs the PCA on them. The literature contains ample evidence [73, 151] that makes the superiority of the ASM over the PCA-based approach abundantly clear. In fact, all the gradient-free methods use the Active Subspace as the benchmark. In the experiment for this question, the subspace that the proposed method computes will be compared with the Active Subspace of identical size using the largest principal subspace angle. Irrespective of whether the subspaces are similar or dissimilar, we argue that the proposed method is effective if the previous experiment is successful because there is no particular reason for the subspaces to be similar. One may argue that the subspaces will indeed be similar because both the ASM and the manifold optimization-based methods explicitly try to impose a condition that maximizes knowledge about the function in a low-dimensional subspace. In chapter 5, a detailed discussion will reveal that the proposed manifold optimization-based method exclusively relies on maximizing the likelihood of observing the training data to find the subspace that captures the function's variation in contrast to ASM's approach of using gradient samples of the function. Albeit the methods are completely different in their formulations, such an outcome will only strengthen the merit of the proposed model and is actually welcomed.

The second part of this question thoroughly evaluates the empirical performance of the method. Performing this exercise is standard for any supervised learning technique. Another purpose is to find out how the method performs in relation to other competing methods. A literature search will establish a recently developed technique [265] as a fair benchmark for comparing the proposed method's relative competitiveness. The benchmark method is similar to the proposed method in that it is gradient-free, works with a small number of samples, and is compatible with unstructured sampling. The outcome of the experiment for this research question is intended to provide practitioners insights into the expected

behavior of the method based on their specific problem. The goal of this experiment is to provide recommendations for questions regarding the number of samples required for a good model that successfully recovers the subspace and its relationship to the number of inputs, number of actual input subspace dimensions etc. This experiment has no specific metric for success. We only hope that our method is at least competitive, if not superior, to the benchmark method and is able to provide useful insights for its intended use with ROMs.

The discussion so far has evaded an important question regarding the existence and size of the low-dimensional input subspace. The ASM has provisions to ascertain whether a low-dimensional subspace exists. If it does, the method also provides a heuristic to fix its size. To claim that the proposed model provides a wholesome solution, it must address this issue at the very minimum. We attempt to answer it using the following question:

Research Question 2.3

Given a function to be approximated using the proposed model, how to assess whether a low dimensional input subspace exists? If it does, how can the dimensionality of the low dimensional input subspace be computed?

The ASM enjoys the advantage of the availability of gradient information to find the subspace. But compatibility with input-output pairs exclusively is a non-negotiable requirement for the proposed method. Chapter 5 will pose the problem of finding the dimensionality of the input subspace as an exercise in model selection. It will be shown how other automatic dimension detection methods used for instance in probabilistic PCA [120], fall short in terms of computational intractability and practical considerations. The literature will be used to propose and justify the use of the Bayesian Information Criterion (BIC) for selecting the dimension of the input subspace. The experiment will be designed to check whether the proposed model performs as expected under the assumptions for faithful use of the BIC when the size of low-dimensional input subspace is known. If successful, the findings of this experiment will be used to propose a heuristic that leverages both the BIC

and the predictive accuracy to check the existence and set the dimensionality of the input subspace.

Once the proposed surrogate model is whetted, we propose a technique to use it for training POD and interpolation-based ROMs. The final question empirically evaluates and quantifies its performance with respect to all the considerations in the previous research questions specifically in the context of constructing predictive ROMs. It reads as follows:

Research Question 2.4

For constructing POD and interpolation-based ROMs, does the application of the developed model to identify a low-dimensional subspace and train surrogates for the map between the POD coefficients and the inputs yield models with good predictive performance at the field output level?

The experiment for this question puts the developed method to test on a canonical field output emulation problem with a large number of inputs. Note that when the developed surrogate model is applied in the pipeline for constructing POD and interpolation-based ROMs, its effectiveness and success will depend on the complexity of the physical features of the problem itself. Several other factors such as the extent of the low-rank structure may also affect the effectiveness of the method. Testing all these considerations is the primary goal of this experiment. If successful, the outcome will first establish whether the method is applicable for ROM construction based on the characteristics of the field output. It is expected that the results will provide recommendations regarding the kind of field emulation problems the method works well with and how a practitioner can go about setting it up for success. Finally, the whole methodology will be applied for training ROMs to emulate the pressure coefficient for two airfoil problems and one 3-dimensional wing problem. We will also provide recommendations and thoughts regarding the developed method's application with other ROM techniques.

3.4 Research Area 3: Feasibility of a Multifidelity Technique for Handling High-Dimensional Inputs and Outputs

Of the two main sources, the second relatively more significant contribution to the *offline* cost is associated with the generation of high-fidelity solutions for constructing ROMs when the underlying analysis is expensive. For practical problems with a high-dimensional parametrization in particular, very expensive analyses for large practical systems give rise to challenging situations where the training dataset is large but not large enough to account for all the field level variations caused by the inputs. In such cases, randomized methods (see Sec. 3.2) can be used for efficient dimension reduction and the surrogate model developed under research area 2 (see Sec. 3.3) can tackle the supervised learning task to encode parametric dependence. As we discussed before however, the lack of sufficient field level data causes the predictive ROMs to be inaccurate i.e. any increase in accuracy must be paid for by incurring a high computational expense to increase the size of the training data.

The discussion in section 1.5.2 presented multifidelity methods as an effective means to address the issue of the conflict between the need for high-accuracy with insufficient data. We also pointed out that studies that deal with multifidelity ROMs are relatively small in number. We will see later that this is mainly because of inconsistencies in the size and topology of the field outputs from analyses differing in fidelities. Another cause for concern is the need to handle large inputs in the reduced representation of the high-fidelity system in a multifidelity context. This research area poses the following overarching question in an attempt to resolve the concerns discussed above:

Research Question 3

In circumstances where the quantity of high-fidelity data is insufficient, how can techniques from multifidelity methods be leveraged to improve the accuracy of parametric ROMs while accommodating a relatively large number of inputs?

The stress on high-dimensional inputs is to ensure that the development stresses on the applicability on practical problems. Another clarification regarding the use of "insufficiency" of data must be clarified. There exists no clear definition of what insufficient means in the

context of constructing ROMs. It largely depends on the characteristics of the problem at hand. In our work, it is assumed, arguably and demonstrably so, that analyses with a large number of inputs and large spatially distributed field quantities often lead to situations where more data is always desirable. Therefore, the focus on equipping the developed method to provide superior predictive accuracy in the presence of additional data from sources inferior in fidelity. The following discussion will highlight the specific challenges that arise when considering inclusion of additional low-fidelity field data in order to improve the predictive accuracy of the ROM when insufficient high-fidelity data is available.

Goal. The goals of the work under this research area are: 1) identification of a suitable nonintrusive parametric multifidelity ROM method that handles field outputs from different sources of data, 2) modification of the identified method to handle a large number of inputs by leveraging and extending the work done in the previous research area to the multifidelity setting, and 3) assessment of the viability of the resulting methodology in its ability to improve predictive accuracy when limited high-fidelity data is available.

The search for the ROM method will focus on accommodation of the largely neglected case where the field outputs from different sources have different sizes. We will later see that most multifidelity ROM methods use sources of data with an identical computational grid; the resulting outputs are of identical size. But as one can imagine, this severely limits the out-of-the-box construction of ROMs when presented with two sets of equivalent analyses set up on different grids (one coarse and the other fine, for instance). It will be shown later that the identified ROM method requires regression of multifidelity low-dimensional representations of the field outputs as a function of a large number of inputs. We propose an extension to the supervised learning model formulated as part of the previous research area in an attempt to address this issue. In particular, the proposed regression technique leverages abundant low-fidelity data and sparse low-fidelity data to improve the accuracy of the discovered low-dimensional input projection subspace and yields a model with superior predictive performance under some assumptions regarding the underlying multifidelity analyses. The success of this effort hinges on the following statement's validity:

Hypothesis 3: *The combination of a multifidelity dimension reduction method that finds a common low-dimensional representation of the field outputs and a multifidelity regression model that aids in the accurate discovery of a low-dimensional structure in the inputs will improve the predictive accuracy of ROMs when limited high-fidelity data is available.*

This statement clearly segregates the effort into finding a ROM method and the development of a regression method. Chapter 6 begins with a brief review of all the multifidelity ROM methods. Then, a recently developed nonintrusive ROM method [195] is identified that can be applied out-of-the-box to field outputs of different sizes based on the aforementioned requirements. It will be shown that the method is similar to the POD and interpolation method in that it performs dimension reduction on the high- and low-fidelity datasets separately. Once the low-dimensional representations are obtained, the method finds a common representation by aligning the individual latent spaces using manifold alignment. In the common latent space thus obtained, a multifidelity regression model is trained to predict the coordinates for an unseen point in the parameter space. However, the resulting method was shown to work only with a small number of inputs because of the use of conventional regression methods.

Recall the specific requirement to handle relatively large input spaces. In an attempt to find an appropriate multifidelity regression technique to accommodate high-dimensional inputs, we perform a brief literature search in chapter 6. The literature search will clearly show that regression with a large number of inputs using mapping-based (see section 2.5) schemes has received limited attention in the multifidelity context. Therefore, we propose an extension to the manifold optimization-based GPR from the previous research area. The proposed method attempts to simultaneously train a multifidelity GPR model and find a "common" low-dimensional input subspace (using a Grassmann manifold parametrization) projection assuming it exists. Whether the proposed method works as expected is tested using the following research question:

Research Question 3.1

Does the proposed manifold optimization-based multifidelity model aid in improving the recovery of the low-dimensional input subspace and the predictive accuracy for the high-fidelity output using additional inexpensive low-fidelity data?

The main purpose of this question is the evaluation of the proposed model's effectiveness in refining the estimate of the low-dimensional input subspace for the high-fidelity function when it is given additional low-fidelity samples. In this formulation, the low-fidelity function's low-dimensional structure is assumed to be either identical or at least similar to that of the high-fidelity function. An obvious and correct suspicion may arise in the reader's mind regarding this assumption. This question is intended to empirically test the ramifications of making such an assumption on the effectiveness of the model. The experiments will be performed on two scalar multifidelity engineering functions. The outcomes will shed light on the performance characteristics of the model in terms of the ratio of high- and low-fidelity samples required to aid in the recovery of the low-dimensional subspace, its sensitivity to the nature of the functions and the effect of the discrepancy between the low-dimensional structures of the high- and the low-fidelity function.

If the proposed model is successful in practice on scalar functions, it will be applied on multifidelity latent space datasets given by the multifidelity manifold alignment-based method. The following research question attempts to investigate whether the proposed multifidelity ROM offers any benefit over the single fidelity POD and interpolation-based ROM:

Research Question 3.2

When additional low-fidelity data is available, does the proposed model leveraging the manifold alignment-based dimension reduction along with the multifidelity manifold optimization-based Gaussian process regression result in an increase in the predictive accuracy relative to the single fidelity POD and interpolation-based model?

The experiment for this question will involve construction of a multifidelity predictive ROM using manifold alignment for dimension reduction and the manifold optimization-based GP for the high-dimensional supervised learning problem in the latent space. Its purpose is to test whether the proposed methodology for handling multifidelity datasets to predict spatially distributed quantities with high-dimensional inputs is effective and successful in practice. The outcome will provide insights into whether a common low-dimensional input structure exists following the alignment of the latent spaces for field output data from different sources. If it does, the results will show if the subspace can be captured by the proposed multifidelity regression model.

Research Areas and Corresponding Chapters. Each chapter that follows consists of the relevant literature to help build the hypothesis for each research question introduced in this chapter under a particular research area. As mentioned earlier, through experiments developed for testing the hypotheses, the goal is to answer the overarching main research question for the research area in question. Figure 19 shows the chapter corresponding to each of the different research thrusts.

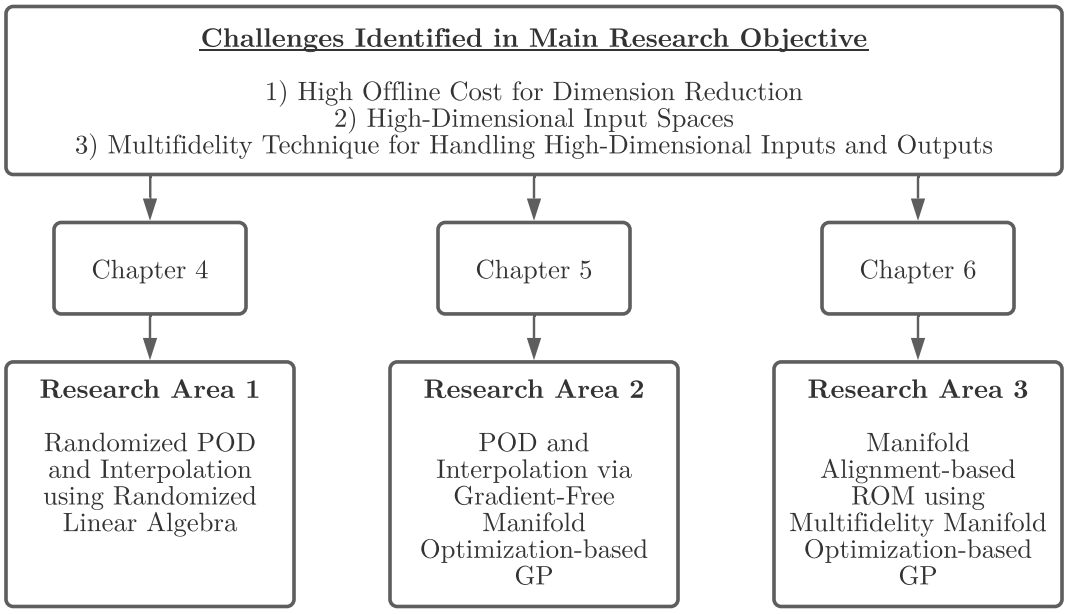


Figure 19: Research Roadmap

CHAPTER IV

POD AND INTERPOLATION USING RANDOMIZED DIMENSION REDUCTION ALGORITHMS

This chapter presents the findings from the work under the first research area i.e. reduction of offline cost associated with the dimension reduction step. Figure 20 shows an overview of the key issue along with an outline of the order of presentation this chapter follows. Let us begin by recollecting that our literature search revealed that the nascent field of randomized linear algebra exploits the idea of randomness to speed-up the computations involved in factorizing a matrix. Some facts and observations regarding computational complexity provided a base for the argument that this strategy is an appealing, if not superior alternative to the *method of snapshots* for the efficiently compressing large snapshot matrices arising from complex, practical engineering analyses. Existing work by researchers [22, 23] for compressing simulation data bolsters the argument for using randomization. However, its effect on the performance of parametric ROMs has received limited attention.

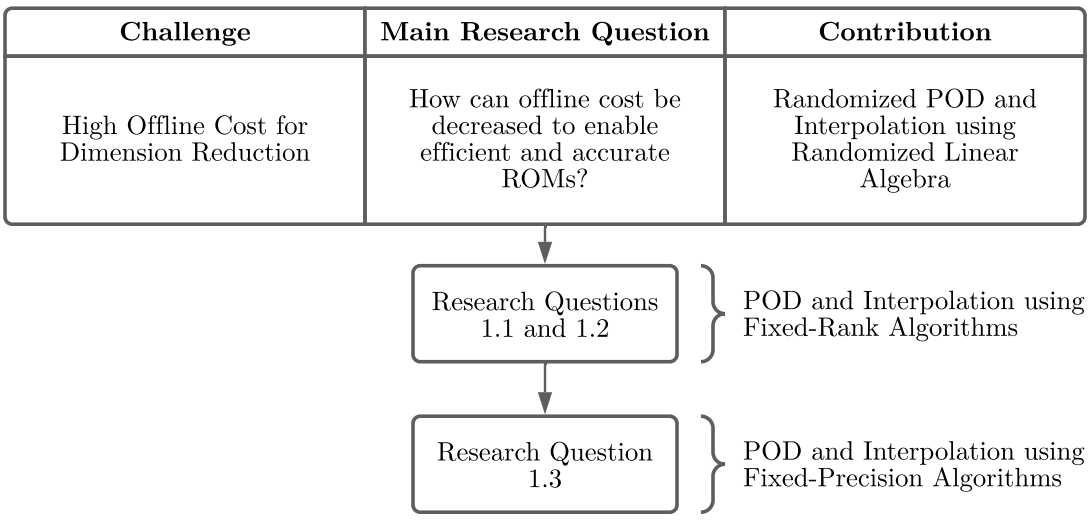


Figure 20: Chapter Outline

We propounded the idea that randomization can indeed be useful for constructing parametric ROMs efficiently and laid out some questions to analyze and investigate its effects in a principled manner. Let us now choose the first set of candidate algorithms to perform our study by understanding how randomized methods compute fast approximate factorizations given a truncation rank.

Remark. The number of repetitions for all the experiments in this chapter was set considering the available computational resources and to reveal any issues regarding bias and/or inconsistency in convergence. Beyond the reported number of repetitions for each experiment, the distribution of the results did not significantly change the observations and conclusions.

4.1 Low-Rank Matrix Approximation using Randomized Linear Algebra

The problem of low-rank matrix approximation appears in many scenarios such as computing directions of maximum variance (nothing other than PCA), embedding high-dimensional data in a low-dimensional space (nothing other than SVD/POD/PCA), solving a system of equations via least-squares to estimate parameters for a linear regression problem, etc. Figure 21 provides a top level view of the concept of exploiting randomness in the context of dimension reduction.

The gamut of methods to decompose matrices typically factorizes an input matrix $\boldsymbol{W}$ into

$$\boldsymbol{W} = \boldsymbol{W}_1 \, \boldsymbol{W}_2, \; \boldsymbol{W} \in \mathbb{R}^{m \times n}, \; \boldsymbol{W}_1 \in \mathbb{R}^{m \times r}, \; \boldsymbol{W}_2 \in \mathbb{R}^{r \times n}$$

where $m, n \in \mathbb{N}$ are the row and column dimensions, and r is the rank of the input matrix X. The primary goal of factorization is to find $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ along with the "numerical" rank r. A numerical rank $r \ll m \;\&\; n$ makes it easy to perform computations with, and store information contained in $\boldsymbol{W}$. In applications involving ROMs, the specific factorization is, of course, the SVD. The snapshot matrix $\boldsymbol{W}$ is the input to the low-rank factorization scheme. Instead of finding the true numerical rank r, the POD (through the SVD) relies

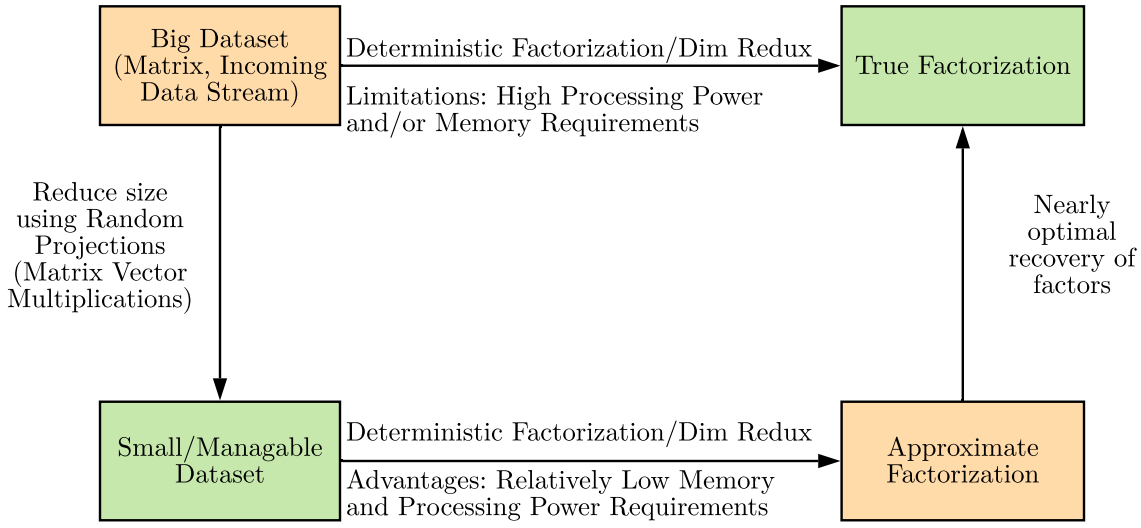


Figure 21: High-Level Idea of Randomized Matrix Factorization Algorithms

on another criterion in the RIC to compute the truncation rank k. In fact, $r \geq k$ if the RIC ≤ 1 with equality being achieved when an RIC of 1 is desired. As one can imagine, randomized formulations for factorization of $\boldsymbol{W}$ rely on and take advantage of the fact that r is typically expected to be much smaller than the matrix dimensions.

As mentioned in passing in chapter 2, randomized techniques use two stages to compute a low-rank approximation for an input matrix. Let us take closer look at the specific goals of the two stages. We follow the explanation in the seminal paper by Halko et. al. [112]. Interested readers are encouraged to refer to their work for more details. The first stage computes an approximate orthonormal basis $\boldsymbol{Q}$ for the range of the input matrix such that

$$\boldsymbol{W} \approx \boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{W}$$

The second stage takes as input matrix $\boldsymbol{Q}$ and uses standard deterministic algorithms to compute the desired factorization of $\boldsymbol{W}$.

In order to facilitate development of the intuition behind randomization, let us briefly state the framework for low-rank approximation. The main concern in low-rank matrix

approximation is to find an orthonormal matrix $\boldsymbol{Q}$ for an input matrix $\boldsymbol{W}$ such that,

$$\|\boldsymbol{W} - \boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{W}\| \leq \epsilon \tag{39}$$

where $\|\cdot\|$ denotes an appropriate matrix norm and ϵ denotes some acceptable error tolerance criterion. It appeals to intuition that the number of columns k in $\boldsymbol{Q} \in \mathbb{R}^{m\times k}$ must be minimized and that its value will depend on the value of both the error tolerance ϵ and the input matrix $\boldsymbol{W}$. Clearly, the optimal way to construct $\boldsymbol{Q}$ is to set it equal to the left singular vectors of $\boldsymbol{W}$. Due to the optimal reconstruction property of the SVD, the value of r can be set such that the $(r+1)^{\text{th}}$ largest singular value is less than ϵ. However, that is precisely what is to be avoided to take advantage of randomization.

4.1.1 Randomization: Intuitive Explanation

Let us first focus on the so-called fixed-rank randomized algorithms where it is assumed that k is known. For reasons to be clarified later, also assume that an additional parameter p called the oversampling parameter is given. The problem of finding a $\boldsymbol{Q} \in \mathbb{R}^{m\times(k+p)}$, that minimizes equation (39) is called the fixed-rank problem. The following examples provide an intuitive insight into how randomness helps in solving the fixed-rank problem for an input matrix $\boldsymbol{W}$ with an actual rank of k. Say, we apply matrix $\boldsymbol{W}$ on k randomly samples vectors $\boldsymbol{\omega}_{(i)} \in \mathbb{R}^{n\times 1},\ i = 1,\dots k$ and collect $\boldsymbol{W}\boldsymbol{\omega}_{(i)},\ i = 1,\dots,k$. Because the sampled vectors $\boldsymbol{\omega}_i,\ i = 1,\dots k$ are random, they are unlikely to be linearly dependent. Consequently, the resulting random matrix-vector products are also likely to be independent. By construction, they span the range of matrix $\boldsymbol{W}$. To satisfy orthonormality of the columns of $\boldsymbol{Q}$, we simply need to orthogonalize the matrix formed by stacking the k random matrix-vector products. The oversampling parameter p just increases the probability of capturing the range of $\boldsymbol{W}$ by using additional samples of $\boldsymbol{\omega}_{(i)} \in \mathbb{R}^{n\times 1},\ i = 1,\dots,(k+p)$. Halko et. al. provide a beautiful explanation for how a small number of additional samples helps. They say, let $\boldsymbol{W}\boldsymbol{\omega}_{(i)} = \boldsymbol{B}\boldsymbol{\omega}_{(i)} + \boldsymbol{E}\boldsymbol{\omega}_{(i)}$, where $\boldsymbol{B}$ contains the information we wish to capture and $\boldsymbol{E}$ is a tiny perturbation. The perturbation can be thought of as an undesired shift to the actual range. But if additional samples of $\boldsymbol{\omega}$ are drawn, the effect of the shift is reduced thereby increasing the chance of capturing the true subspace.

Of the two stages in the computation of randomized factorization, the first stage involving matrix multiplication typically dominates the second stage in terms of computational cost. Even though one may wish to compare this cost to the matrix multiplication cost incurred in the method of snapshots, it must be stressed that one of the real powers of randomization is the possibility of arranging and manipulating the matrix multiplication to attain maximum efficiency.

4.1.2 Randomized Singular Value Decomposition

The first randomized POD algorithm that is tested as a suitable candidate for performing dimension reduction in the context of ROMs is called the Randomized Singular Value Decomposition (rSVD). It is a fixed-rank algorithm that naturally follows from the two stage scheme discussed above. Algorithm 2 describes the steps for computing a rank-k approximation of an input matrix $\boldsymbol{W}$ using the rSVD. Note that the reduction in the size occurs

Algorithm 2: The Randomized Singular Value Decomposition Algorithm

Input: Matrix $\boldsymbol{W} \in \mathbb{R}^{m \times n}$, target rank ($k$), over sampling parameter (p), number of power iterations (q)

Output: Approx. $\boldsymbol{U} \in \mathbb{R}^{m \times k}$, approx. $\boldsymbol{\Sigma} \in \mathbb{R}^{k \times k}$, and approx. $\boldsymbol{V} \in \mathbb{R}^{n \times k}$

1 Generate sampling matrix $\boldsymbol{\Omega} \in \mathbb{R}^{n \times (k+p)}$

2 Compute $\boldsymbol{Y} = (\boldsymbol{W}\boldsymbol{W}^T)^q \boldsymbol{W}\boldsymbol{\Omega}$ by pre-multiplying $\boldsymbol{W}\boldsymbol{\Omega}$ alternately with $\boldsymbol{W}$ and $\boldsymbol{W}^T$ `// power iterations`

3 Compute orthonormal matrix $\boldsymbol{Q}$ spanning the range of $\boldsymbol{Y}$

4 Compute $\boldsymbol{B} = \boldsymbol{Q}^T\boldsymbol{W}$ `// note the reduction in size of the input matrix`

5 Compute SVD for $\boldsymbol{B} = \tilde{\boldsymbol{U}}\boldsymbol{\Sigma}\boldsymbol{V}^T$ Assign $\boldsymbol{U} = \boldsymbol{Q}\tilde{\boldsymbol{U}}$ Return $\boldsymbol{U}$, $\boldsymbol{\Sigma}$, $\boldsymbol{V}$

at step 3 through the action of the input matrix on the random sampling matrix $\boldsymbol{\Omega}$. The sampling matrix randomly samples the column space of the matrix $\boldsymbol{A}$ to form matrix $\boldsymbol{Y}$ which is much smaller in size and therefore computationally inexpensive to factorize. The basic idea is that if $\boldsymbol{\Omega}$ captures sufficient information from the input matrix, $\boldsymbol{Y}$ can provide an approximation of the SVD of $\boldsymbol{A}$. Several algorithms and options exist for generating the random sampling matrix $\boldsymbol{\Omega}$ [150]. This work utilizes the most basic dimension reduction map i.e. a Gaussian matrix with individual entries drawn from independent standard normal distributions. A Gaussian map $\boldsymbol{\Omega} \in \mathbb{R}^{a \times b}$ requires $a \times b$ storage and its application on a vector costs $\mathcal{O}(ab)$ floating point operations. Figure 22 provides a graphical depiction of the

rSVD method. As discussed above, the oversampling parameter p increases the probability

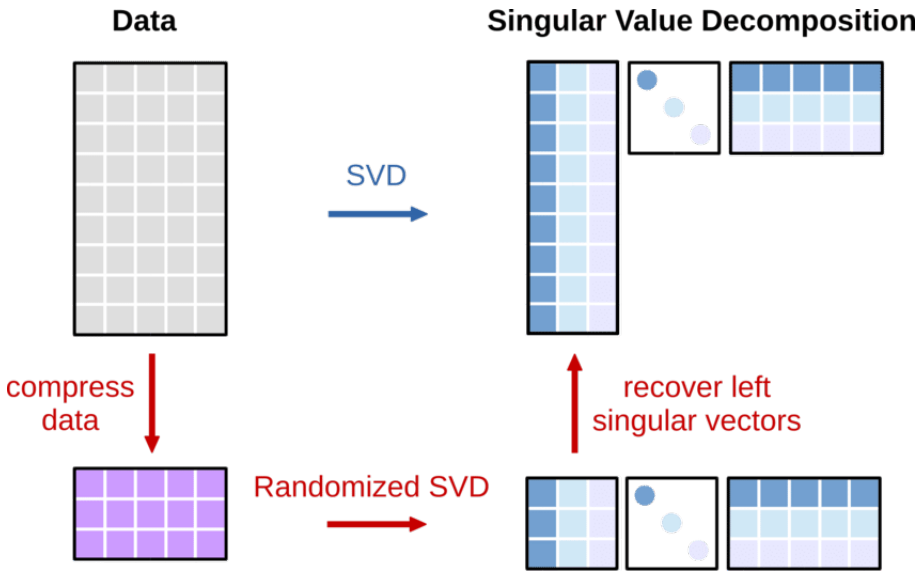


Figure 22: Graphical Depiction of the Randomized SVD (from [89])

of capturing the range of $\boldsymbol{W}$. However, notice that the algorithm requires as input an additional parameter in q denoting the number of power/subspace iterations. q also controls the error of the approximate factorization. Succinctly put, a small q improves the accuracy of the approximation for matrices with a slowly decaying singular value spectrum while a large p increases the odds of achieving oracle approximation error.

To understand how q actually helps in improving the approximation, it is instructive to understand the odds with which randomization computes accurate low-rank approximations through an elegant theorem from [112] restated here,

Theorem. *Suppose that $\boldsymbol{W}$ is a real matrix of size $m \times n$. Select an exponent q and a target number of k singular values such that $2 \leq k \leq 0.5\ min(m, n)$. Execute the Randomized SVD algorithm to obtain a rank-$2k$ factorization $\boldsymbol{U\Sigma V}^T$. Then*

$$\mathbb{E}\,\|\boldsymbol{W} - \boldsymbol{U\Sigma V}^*\| \leq \left[1 + 4\sqrt{\frac{2\min\{m,n\}}{k-1}}\right]^{1/(2q+1)} \sigma_{k+1} \tag{40}$$

where $\mathbb{E}$ is the expectation operator with respect to the random test matrix and σ_{k+1} is the $(k+1)^{\text{th}}$ singular value of matrix $\boldsymbol{W}$.

This powerful result shows that the randomized scheme computes basis vectors that are

within a small polynomial factor of the oracle minimum. Recall that the best possible approximation error is given by σ_{k+1}. Especially in our case, since we are concerned with a rank-k truncation, the error bound becomes more favorable by giving

$$\mathbb{E}\left\|\boldsymbol{W} - \boldsymbol{U}\boldsymbol{\Sigma}_{(k)}\boldsymbol{V}^*\right\| \leq \sigma_{k+1} + \left[1 + 4\sqrt{\frac{2\min\{m,n\}}{k-1}}\right]^{1/(2q+1)} \sigma_{k+1} \tag{41}$$

where $\boldsymbol{\Sigma}_k$ is the diagonal matrix containing only the k-largest singular values. First, note that a small q decreases the value of the polynomial factor that decreases the worst case error in the approximation on average. Although power iterations do not significantly benefit matrices that have a sharp singular value decay, they are extremely useful for matrices that are either too large or have a slowly decaying singular value spectrum or both. In short, the interference of small singular values on the randomized calculations is decreased by taking powers of the input matrix. Say, instead of directly sampling $\boldsymbol{W}$, the matrix $(\boldsymbol{W}\boldsymbol{W}^T)^q\boldsymbol{W}\boldsymbol{\Omega}$ is sampled. Even though $(\boldsymbol{W}\boldsymbol{W}^T)^q\boldsymbol{W}$ has the same left and right singular vectors, the power iterations make its singular values decay faster because $\sigma_i\left((\boldsymbol{W}\boldsymbol{W}^T)^q\boldsymbol{W}\right) = \sigma_i(\boldsymbol{W})^{2q+1}$. In essence, the minuscule singular values disappear as q increases. Despite the overhead of $(2q+1)$ additional matrix multiplication, power iterations exponentially benefit the approximation error for large matrices with a flat singular value spectrum.

4.1.3 Randomized Sketching-Based Singular Value Decomposition

Another line of work bearing semblance to compressed sensing is referred to as matrix sketching-based algorithms [89]. These algorithms randomly extract specific portions of the input matrix called *sketches* to construct a low-rank approximation. They also enjoy the provision of a posteriori error estimates for the approximate factorization from the rich theory of compressed sensing and random linear algebra. As one can imagine, these methods can be extremely useful in memory constrained situations. They are similar in nature to randomized methods in that the low-rank approximation is inexact.

The second promising candidate algorithm belongs to this class of randomized algorithms. Tropp et. al. [252] very recently developed a novel state-of-the-art sketching-based scheme especially for large-scale scientific simulation data. Their specific focus on "streaming" applications perfectly aligns with the aforementioned concerns regarding issues related

to storage and processing of large snapshot matrices. It must be noted that their algorithm focuses on reducing the memory requirements while at least maintaining, if not reducing the time complexity offered by similar randomized schemes. Algorithm development efforts by the original authors focused on minimization of the approximation error while also ensuring a reduction in the amount of storage required. The focus of this work will be on thoroughly evaluating its viability for use with parametric ROMs. What follows is a brief introduction of the basic idea of the method - interpreted, adapted, and reproduced from [251, 252].

The sketching-based randomized SVD is a fixed-rank algorithm that computes a rank-k approximation of an input matrix $\boldsymbol{W} \in \mathbb{R}^{m \times n}$ using four random linear maps parametrized by the so-called *range* parameter r and *core* parameter p such that

$$k \leq r \leq p \leq \min\{m, n\} \tag{42}$$

The four independently drawn linear reduction maps are

$$\begin{aligned} \boldsymbol{\Upsilon} \in \mathbb{R}^{r \times m} \quad \text{and} \quad \boldsymbol{\Omega} \in \mathbb{R}^{r \times n}, \\ \boldsymbol{\Phi} \in \mathbb{R}^{p \times m} \quad \text{and} \quad \boldsymbol{\Psi} \in \mathbb{R}^{p \times n} \end{aligned} \tag{43}$$

The method then proceeds to compute three "sketches"

$$\begin{aligned} \boldsymbol{X} := \boldsymbol{\Upsilon}\boldsymbol{W} \in \mathbb{R}^{r \times n} \text{ and } \boldsymbol{Y} := \boldsymbol{W}\boldsymbol{\Omega}^T \in \mathbb{R}^{m \times r}, \\ \boldsymbol{Z} := \boldsymbol{\Phi}\boldsymbol{W}\boldsymbol{\Psi}^T \in \mathbb{R}^{p \times p} \end{aligned} \tag{44}$$

The idea of forming and leveraging matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ to capture the row space (corange) and column space (range) of $\boldsymbol{W}$ is not new. In fact, the rSVD algorithm precisely uses a random matrix to extract the range of $\boldsymbol{W}$. However, the authors in [252] assert and demonstrate that the new sketch $\boldsymbol{Z}$ named the core sketch contains potentially unused useful information that improves the estimates of the singular values and vectors of $\boldsymbol{W}$. Note that the strength of this method lies in the fact that all further computations exclusively use the sketches. The answers to how much the factorization is computationally efficient, accurate, and the trade-offs involved thereof are purely decided by the choice of random matrices in equation (43), the parameters $r, k,$ and, p, and the characteristics of the input matrix $\boldsymbol{W}$.

Let us now look at a simple explanation for how the sketching-based approach is superior to contemporary methods. The intuition behind capturing information in $\boldsymbol{W}$ using random

matrix-matrix products remains identical to the discussion for the rSVD. For computing a low-rank approximation of $\boldsymbol{W}$, this method uses orthonormal bases $\boldsymbol{P}$ and $\boldsymbol{Q}$ of $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively, obtained using a thin `QR` decomposition and the core sketch $\boldsymbol{Z}$ to form what is called the core approximation $\boldsymbol{C}$ of $\boldsymbol{W}$, given by

$$\boldsymbol{C} := (\boldsymbol{\Phi Q})^{\dagger} \boldsymbol{Z} \left((\boldsymbol{\Psi P})^{\dagger} \right)^{T} \in \mathbb{R}^{r \times r} \tag{45}$$

The core approximation contains additional information about the action of $\boldsymbol{W}$ between spaces spanned by $\boldsymbol{Q}$ and $\boldsymbol{P}$. It is used to construct an approximation of $\boldsymbol{W}$, denoted here as $\hat{\boldsymbol{W}}$, given by

$$\hat{\boldsymbol{W}} := \boldsymbol{QCP}^{T} \tag{46}$$

The intuition behind equation 45 is as follows: 1) Recall that $\boldsymbol{P}$ and $\boldsymbol{Q}$ capture the corange and range of $\boldsymbol{W}$ and therefore it is reasonable to say that

$$\boldsymbol{W} \approx \boldsymbol{Q}(\boldsymbol{Q}^{T}\boldsymbol{WP})\boldsymbol{P}^{T} \tag{47}$$

2) since the term within the brackets cannot be formed explicitly because its computation depends on $\boldsymbol{W}$, the core sketch can be used instead as

$$\boldsymbol{C} = (\boldsymbol{\Phi Q})^{\dagger} \boldsymbol{Z} \left((\boldsymbol{\Psi P})^{\dagger} \right)^{*} \approx \boldsymbol{Q}^{T}\boldsymbol{WP} \tag{48}$$

3) Using the relations in equations (47) and (48), we can write

$$\boldsymbol{W} \approx \boldsymbol{Q} \left(\boldsymbol{Q}^{T}\boldsymbol{WP} \right) \boldsymbol{P}^{T} \approx \boldsymbol{QCP}^{T} = \hat{\boldsymbol{W}} \tag{49}$$

A salient feature of this algorithm is that the underlying theory suggests values for the algorithm-specific parameters (the sizes of the sketches) such that a rank$-k$ approximation is optimal for a given storage budget parametrized by the real number s called the storage factor. The algorithm computes the optimal values of p and r for a given q and storage budget $T = s \times (m + n)$. The following theorem give the error bound for the approximation $\hat{\boldsymbol{W}} := \boldsymbol{QCP}^{T}$:

Theorem (see [252] for proof). *Suppose that $\boldsymbol{W}$ in a real matrix of size $m \times n$. Assume that the sketch parameters satisfy $p \geq 2r+1$. For sketches $\boldsymbol{X}, \boldsymbol{Y},$ and $\boldsymbol{Z}$ of $\boldsymbol{W}$ formed using*

independently drawn Gaussian matrices $\boldsymbol{\Upsilon}, \boldsymbol{\Omega}, \boldsymbol{\Phi}$, *and* $\boldsymbol{\Psi}$, *the rank-k approximation of* $\boldsymbol{W}$ *satisfies*

$$\mathbb{E}\|\boldsymbol{W} - [\hat{\boldsymbol{W}}]_k\|_2 \leq \tau_{k+1}(\boldsymbol{W}) + 2\left[\frac{p-1}{p-r-1} \cdot \min_{e<r-1} \frac{r+\varrho-1}{r-\varrho-1} \cdot \tau^2_{\varrho+1}(\boldsymbol{W})\right]^{1/2} \tag{50}$$

This powerful and optimistic result shows that the error bound can be controlled by manipulating factors which are functions of the algorithm specific parameters in equation (50). Such a manipulation using an optimization problem leads to the following equations for setting r and p, given a storage budget T:

$$\begin{aligned} r_* &:= \left\lfloor \tfrac{1}{8}\left(\sqrt{(m+n+4)^2 + 16\,(T-1)} - (m+n+4)\right)\right\rfloor \\ p_* &:= \left\lfloor \sqrt{T - r_*(m+n)}\right\rfloor \end{aligned} \tag{51}$$

Algorithm 3 presents the detailed steps of the sketching-based randomized algorithm for computing a rank-k approximation of an input matrix $\boldsymbol{A}$ for affordable storage budget T $= s \times (m+n)$ where s is referred to as the storage factor. In this work, the three sketches (randomly sampled matrices $\boldsymbol{X}, \boldsymbol{Y}$ and $\boldsymbol{Z}$) to capture the *range* and *corange* of the input matrix $\boldsymbol{W}$ are formed using four random Gaussian sampling matrices $\boldsymbol{\Upsilon}, \boldsymbol{\Omega}, \boldsymbol{\Phi}$, and $\boldsymbol{\Psi}$. The Gaussian matrices contain entries drawn from independent standard normal distributions as discussed previously for the rSVD.

Algorithm 3: Sketching-Based Randomized Singular Value Decomposition

Input: Matrix $\boldsymbol{W} \in \mathbb{R}^{m\times n}$, target rank ($k$), storage factor ($s$)
Output: Approx. $\boldsymbol{U} \in \mathbb{R}^{m\times k}$, approx. $\boldsymbol{\Sigma} \in \mathbb{R}^{k\times k}$, and approx. $\boldsymbol{V} \in \mathbb{R}^{n\times k}$

1 Use s to compute r and p (see eq. (51)) to construct sampling matrices $\boldsymbol{\Upsilon} \in \mathbb{R}^{r\times m}$, $\boldsymbol{\Omega} \in \mathbb{R}^{r\times n}$, $\boldsymbol{\Phi} \in \mathbb{R}^{p\times m}$, $\boldsymbol{\Psi} \in \mathbb{R}^{p\times n}$
2 Compute orthonormal matrix $\boldsymbol{Q}$ spanning sketch $\boldsymbol{Y} = \boldsymbol{W}\boldsymbol{\Omega}^T$
3 Compute orthonormal matrix $\boldsymbol{P}$ spanning sketch $\boldsymbol{X}^T$, given by $\boldsymbol{X} = \boldsymbol{\Upsilon}\boldsymbol{W}$
4 Solve two small least-square problems to get $\boldsymbol{C} = (\boldsymbol{\Phi}\boldsymbol{Q} \setminus \boldsymbol{Z})/((\boldsymbol{\Psi}\boldsymbol{P})^T)$ where sketch $\boldsymbol{Z} = \boldsymbol{\Phi}\boldsymbol{A}\boldsymbol{\Psi}^T$
5 Compute SVD for $\boldsymbol{C} = \tilde{\boldsymbol{U}}\boldsymbol{\Sigma}\boldsymbol{V}^T$ and truncate to rank-k
6 Set $\boldsymbol{U} = \boldsymbol{Q}\boldsymbol{U}$
7 Set $\boldsymbol{V} = \boldsymbol{P}\boldsymbol{V}$
8 Return $\boldsymbol{U}$, $\boldsymbol{\Sigma}$, $\boldsymbol{V}$

This section has identified and introduced the rSVD and the sketching-based rSVD as the two candidate randomized algorithms to perform dimension reduction. Before formulating the accompanying hypothesis, two obvious but important facts must be restated.

Remark 1. Both the algorithms shown above incur some loss in accuracy. However, in practice, they work well and have controllable bounds on the error due to randomness as we have attempted to explain using some intuition.

Remark 2. As mentioned before, sampling matrices are constructed by independent and identically distributed (i.i.d.) samples from a standard Gaussian [150]. Faster algorithms (not considered) exist for structured sampling matrices but their use is deferred to future work because the aim of this work is to test the viability of randomization as a strategy and to demonstrate some time savings, not necessarily the best possible time savings.

4.2 Quality of POD Basis Obtained by Randomized Algorithms

We are now prepared to build and test the hypothesis for the first research question under this research area:

Research Question 1.1

Can randomized variants of the SVD yield basis vectors that are suitable for construction of parametric ROMs?

In the previous section, we noted how both the candidate randomized algorithms for computing the POD via the SVD produce approximate basis vectors. The quality of the approximate singular values and vectors, the probability of achieving the error bound, and the computational savings depend on the following:

1. the input matrix of snapshots itself, for both the rSVD and the sketching-based rSVD, specifically its actual singular value spectrum. In general, the theorems suggest that matrices with a rapid decay are easy to factorize as opposed to matrices that exhibit a slow decay.

2. the truncation rank k.

3. for the rSVD algorithm, the number of additional samples p and the number of subspace/power iterations q have a significant effect on the computed factorization.

4. for the sketching-based rSVD algorithm, the core parameter p and the range parameter r decide how to achieve the best approximation given the amount of affordable storage T.

The first step in deciding whether these algorithms are competitive and can replace deterministic algorithms depends on their ability to produce singular vectors that are comparable to the ones produced by the deterministic SVD. Because their adoption would also involve setting some algorithm specific parameters, it is important to understand how the sensitivity of the computed approximate factors affects the relationship (or discrepancy) between the randomized and deterministic basis vectors. As a first step therefore, given a known truncation rank, we seek to understand the effect of the algorithm specific parameters on the computed approximate factors for the two chosen fixed-rank random algorithms and compare them to the ones obtained by the truncated SVD algorithm. Now, since the randomized algorithms try to minimize the error due to randomization by design, it is reasonable to state the following hypothesis:

Hypothesis 1.1: *If the basis vectors computed by randomized variants of the SVD are either sufficiently accurate in comparison to the ones computed by the deterministic algorithm or can be manipulated by algorithm specific parameters to achieve comparable accuracy, randomized techniques can enable efficient (as measured by time complexity) computation of approximate basis vectors suitable for parametric ROMs.*

The quantification of the discrepancy between the basis sets obtained from the randomized and deterministic algorithms must be evaluated using objective metrics that are sensitive to the effect of algorithm-specific parameters. In this experiment, the discrepancy between subspaces is measured using the principal angles between subspaces [130]. Given two basis sets corresponding to $\boldsymbol{\Phi}_{k1} \in \mathbb{R}^{m\times k_1}$ and $\boldsymbol{\Phi}_{k2} \in \mathbb{R}^{m\times k_2}$, the cosines of the angles between subspaces can be obtained by first forming

$$\boldsymbol{S} = \boldsymbol{\Phi}_{k1}^T \boldsymbol{\Phi}_{k2}, \quad \boldsymbol{S} \in \mathbb{R}^{k1\times k2} \tag{52}$$

Then, the singular values of $\boldsymbol{S}$ give the cosines of the principal angles. The results in this

work report the angles obtained by taking the inverse of the sine of the largest angle instead of the cosine because of its superior numerical accuracy [130]. The largest principal angle between the rank-k POD basis computed by the randomized algorithm for a specific set of parameters and the deterministic algorithm serves as a measure of discrepancy and is the first metric for this experiment.

Another objective metric to quantify the quality of basis $\boldsymbol{\Phi}_k$ is the relative reconstruction error of the snapshot matrix due to the POD-basis given by

$$\text{Rel. Reconstruction Error} = \frac{\|\boldsymbol{W} - \boldsymbol{\Phi}_k \boldsymbol{\Phi}_k^T \boldsymbol{W}\|_F}{\|\boldsymbol{W}\|_F} \tag{53}$$

Note that equation (53) is a normalized version of the objective (in the Frobenius-norm) that low-rank matrix approximation algorithms minimize. By definition, when evaluated with the basis sets obtained by randomized algorithms, its value must be larger than its oracle value yielded by the true deterministic rank-k basis. Therefore, there is no value in comparing its value between the deterministic and randomized algorithms. Instead, since smaller values of the relative reconstruction error imply higher accuracy of the low-rank approximation, this experiment merely uses this metric to quantify the effect of the additional parameters required by the randomized algorithms. As an example, for the low-rank approximation of some matrix $\boldsymbol{W}$ using the rSVD, it may indicate when the reconstruction error stops reducing as the number of additional samples is increased. Then, the smallest possible approximate reconstruction error when compared with the actual reconstruction error may give an insight into the best possible factorization that the rSVD can provide along with an estimate of the additional computational time penalty.

Canonical Test Problem Description. This experiment is first performed on a non-linear and parametric canonical problem used as a candidate in ROM-based studies [64]. All the experiments in this research area and the dissertation are first demonstrated on canonical problems that either possess the characteristics of a practical problem or can be manipulated to control the severity of a given trait. One may wonder why this is the case given the discussion so far has stressed on practical problems particularly. In some experiments, it is important to be able to study the effects of varying certain aspects of the

problem on the outcome of the experiment. For instance, as we will see later in the case of measuring the wall-time for randomized algorithms, it is rather convenient to change the size of the solution for a canonical problem rather than a practical problem because 1) practical problems are typically more expensive to solve and therefore cost a significant overhead to collect data and 2) it is not a trivial task to setup geometries of varying sizes for practical problems to create snapshots of different sizes.

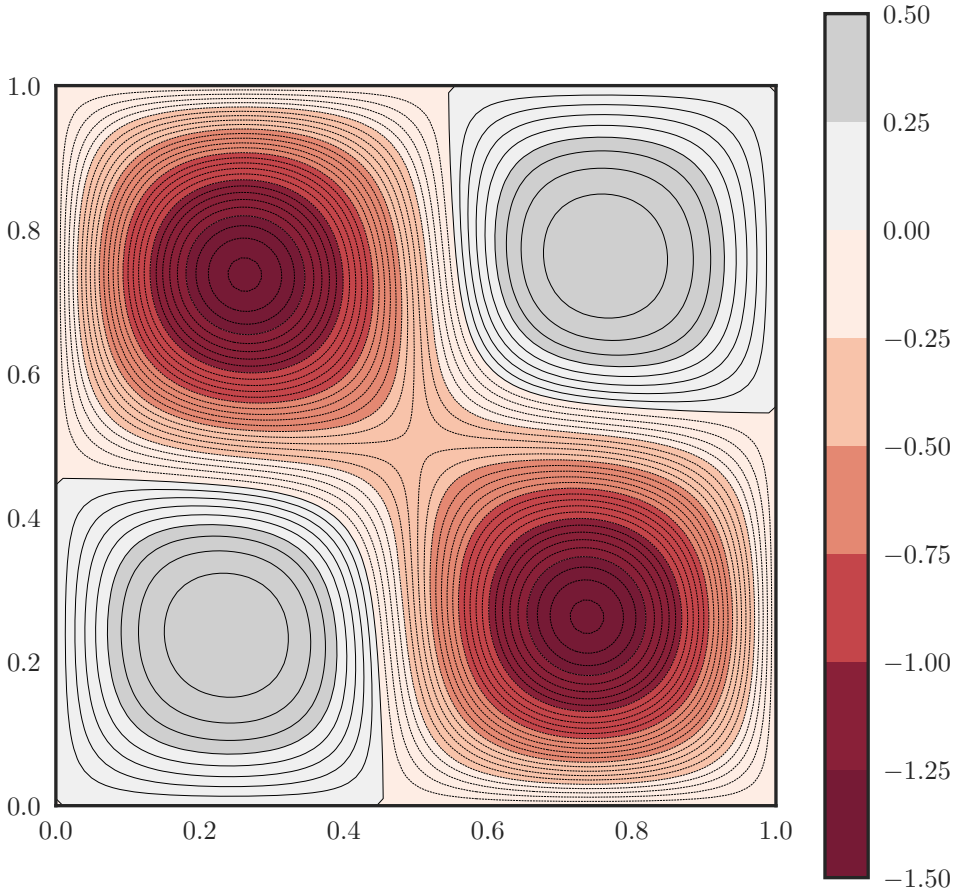


Figure 23: Solution for the Parametric 2D Nonlinear Diffusion Equation at $\boldsymbol{\mu} = [10, 10]$.

Before proceeding with the results of the experiments, we furnish relevant details on the canonical problem for this research area. The first numerical example considered in this work is a nonlinear parametric elliptic PDE that closely follows the work in [64] and is given

in its non-discretized form by

$$-\nabla^2 w + s(w(x,y);\ \boldsymbol{\mu}) = f(x,y), \quad (x,y) \in \boldsymbol{\Omega} \tag{54}$$

$$w(x,y) = 0, \quad (x,y) \in \partial\boldsymbol{\Omega} \tag{55}$$

where $\boldsymbol{\mu} : [\mu_1, \mu_2]^T = [0.01, 10.0]^2 \in \mathbb{R}^2$ denotes the parameter set, w denotes continuous form the field variable, $(x,y) \in \boldsymbol{\Omega} = (0,1)^2$ denotes the 2-D Euclidean space, and the source and nonlinearity are given by

$$f(x,y) = 100\ \sin(2\pi x)\ \sin(2\pi y) \tag{56}$$

$$s(\boldsymbol{\mu}, w) = \frac{\mu_1}{\mu_2}(e^{\mu_2 w} - 1) \tag{57}$$

The nonlinear equations resulting from the finite-differences in space are solved using Newton's method implemented in the Python programming language. The size of the unknowns or dimensionality of the field is $m = 10000$. The snapshots ($n = 500$) are obtained using a Latin-hypercube sampling (LHS) DoE. Figure 23 shows the computational grid along with the visualization of one sample solution.

Details of the Experiment. The experiment begins with the generation and collection of snapshots for some space filling DoE (LHS in this case) by evaluating the expensive full order model. Then, the deterministic SVD is computed to determine the actual truncation rank that satisfies a prescribed RIC. Using the known truncation rank k, appropriate values of the algorithm specific parameters for both the randomized SVD and the sketching-based randomized SVD are chosen such that the whole spectrum of sensible values is covered.

Finally, the randomized factorization is performed for each of the algorithm-specific parameter values (see Table 3) followed by the evaluation and collection of the largest principal subspace angle and the relative reconstruction error. Since the randomized algorithms are stochastic by nature, results for the SVD computations are reported as observed over 20 trials for each method. Figure 24 shows a step-by-step walk through for this experiment.

Results. As evident from figure 25, this particular problem shows strong low-rank structure in the input parameter range. The RIC approaches 100% within the first 10

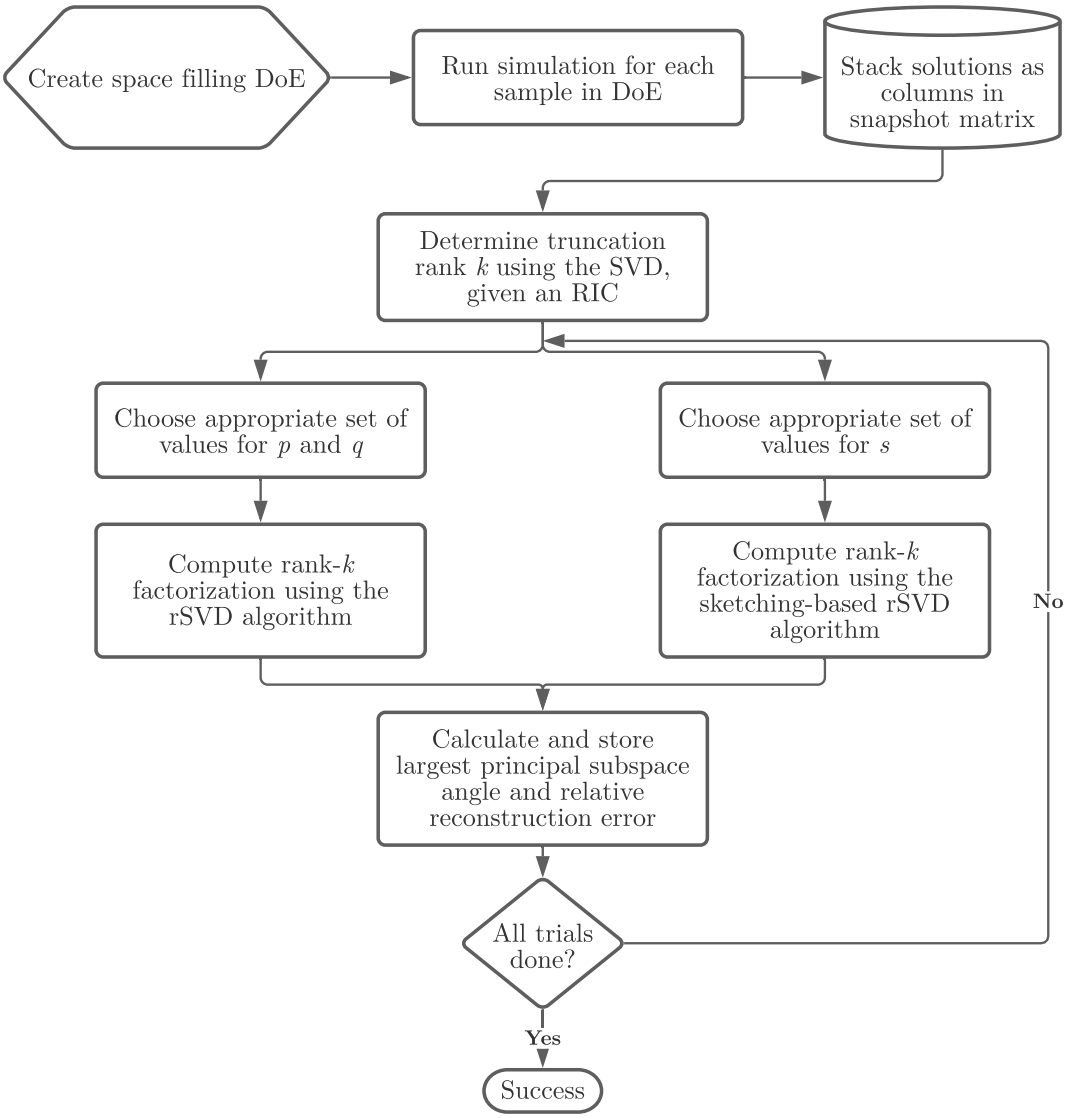


Figure 24: Setup for Experiment 1.1.

Table 3: Dimension Reduction Algorithms for Computing the POD basis

Dimension Reduction Variant	**Algorithm Parameters**
Deterministic SVD (SciPy SVD)	k
Randomized SVD (Rand SVD)	k, p, and q
Sketching-based Randomized (Rand Sketch SVD)	k, s

modes. Therefore, the truncation rank is chosen as $k = 6$. Note that this example is a prime candidate for ROMs due to its extremely sharp singular value decay spectrum as a

result of the low-rank structure.

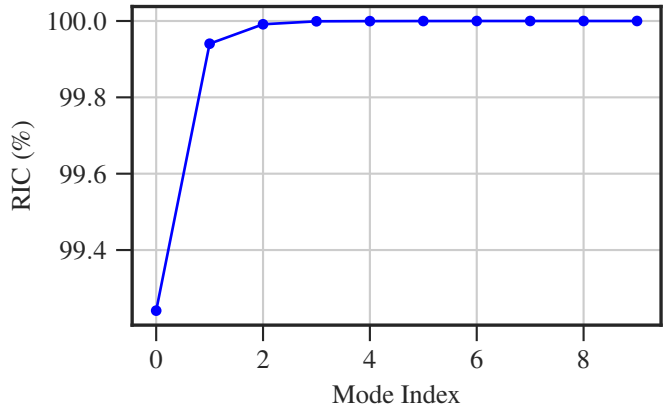
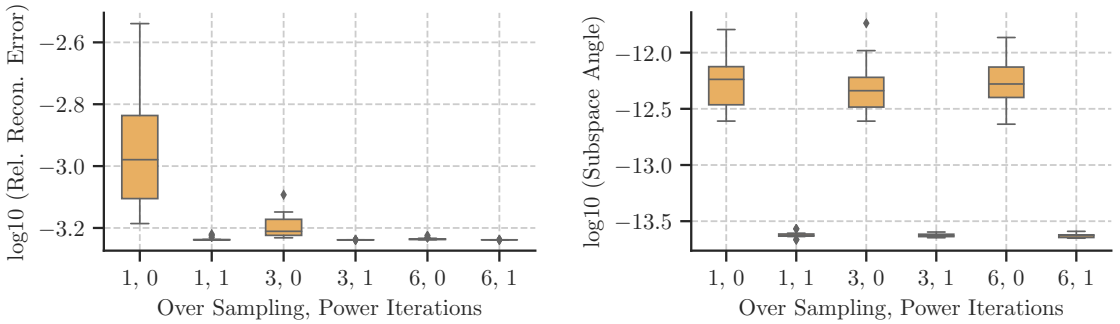


Figure 25: RIC for the Canonical Problem

For the randomized SVD, the sensitivity of a rank-6 approximation on the over sampling parameter and the number of power iterations over 50 trials is shown in figure 26. It can be observed that the number of power iterations has a stronger influence than the over sampling parameter on the relative reconstruction error. Because the number of power iterations provides a mechanism for achieving a faster decay for systems with a slowly decaying spectrum, it is expected to have a strong influence for systems with a fast decay to begin with, as is the case for the canonical problem. As the over sampling parameter increases up to the approximation rank, the method yields subspaces with lower reconstruction error as expected. The typical recommendation is to set the over sampling parameter to a small constant to improve the quality of the approximation. In general, the smaller the over sampling parameter for a given rank k, the cheaper it is to draw the $n \times (k + l)$ sampling matrix. For this problem, observe that even 1 additional sample and 1 power iteration give a reconstruction error that is competitive with the deterministic SVD. Additional sampling beyond half the truncation rank provides only diminishing benefits. The principal subspace angles between the deterministic POD basis and randomized POD basis are near zero across the range of algorithm parameters. Perhaps due to the low rank structure of the snapshot matrix, the randomized SVD algorithm is able to accurately compute the subspace relatively easily. Note the acute effect of the oversampling parameter on the spread of the already negligible angle. This is consistent with our observation regarding its effect on the

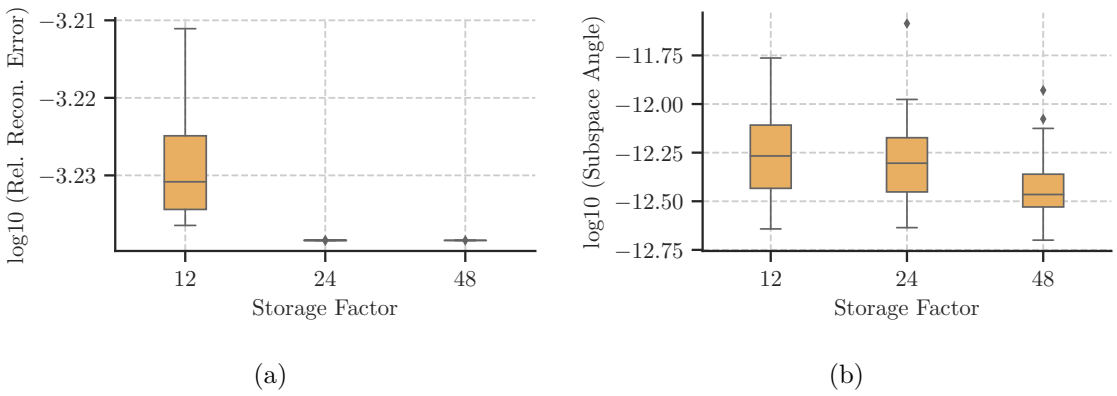
reconstruction error, in turn also reinforcing the fact that a lower subspace angle implies a lower reconstruction error. Owing to the high accuracy of the randomized SVD bases across the range of algorithm parameters, any setting may be chosen to create ROMs for this problem. In this study, the oversampling parameter and power iteration parameters are chosen as 3 and 1, respectively.



(a) (b)

Figure 26: Randomized SVD: Sensitivity of Projection Error and Quality of Basis for the Canonical Problem

Figure 27 shows the sensitivity of the quality of basis approximation on the algorithm-specific parameters for the sketching-based randomized algorithm. It is worth repeating that although the sketching-based algorithm has a larger number of native parameters, their specific choice is obtained through an optimization procedure for a given storage budget specified using the storage factor s. The storage budget T is computed as $T = s \times (m+n)$, where s is the storage factor and m and n are the row and column dimensions of the matrix. It can be seen that the accuracy of the approximation increases as the affordable storage increases. The range of subspace angle values is similar to that of the randomized SVD. Perhaps a higher spread in the values occurs because of a larger compression of the actual dataset due to the relatively higher number of random matrix-vector multiplications to extract the sketch. Note that both the expected value and the range of the distribution of subspace angles decrease at higher storage values. In essence, because of good overall accuracy, any storage factor should be appropriate for creating a ROM.



(a) (b)

Figure 27: Sketching-Based Randomized SVD: Sensitivity of Projection Error and Quality of Basis for the Canonical Problem

Summary. Figures 26 and 27 show promising results regarding the quality of the randomized basis vectors. The following qualitative observations summarize the findings and support the hypothesis. One, the reconstruction error stops decreasing at relatively small values of the algorithm-specific parameters for both the randomized fixed-rank dimension reduction algorithms. The accompanying near-zero subspace angles also indicate that this reconstruction error is in fact nearly identical to the ones achieved by the deterministic algorithm. Two, both the reconstruction error and the subspace angles are consistently achieved within a narrow band of uncertainty at larger values of the algorithm-specific parameters as dictated by the theorems presented above. This implies that repetitions need not be performed if the algorithm-specific parameters are set at relatively higher values. Three, power iterations have a significantly larger effect on the quality of the basis vectors. However, they do come at the price of a higher computational cost. Based on the findings for this test problem, the fixed-rank randomized schemes are suitable for construction of parametric ROMs when computed with the correct settings for the algorithm specific parameters as was hypothesized.

4.3 POD and Interpolation-Based ROMs using Fixed-Rank Randomized Algorithms

The previous experiment gave a strong indication that randomization has the potential to yield basis vectors that are similar to deterministic ones, at least for the chosen candidate problem. The next research question is concerned with actually evaluating whether these basis vectors yield models that are competitive with the deterministic POD and interpolation-based ROMs.

Research Question 1.2

Are parametric ROMs constructed using randomized algorithms for the SVD step competitive in terms of predictive accuracy when compared to ROMs constructed using the deterministic SVD algorithm?

The formulation of the hypothesis for this research question is rather straightforward. Given that the algorithm-specific parameters are set such that we obtain sufficiently accurate basis vectors, it stands to reason that the coordinates obtained by projecting the snapshots onto both the approximate and the actual POD subspace should be similar in value and therefore must have identical variation with respect to the design variables/input parameters. The surrogate models to be trained in the latent space will also be similar. Consequently, the resulting ROMs will have comparable predictive accuracies. Stated as a hypothesis, we can say:

Hypothesis 1.2: *If the basis vectors produced by the randomized algorithms are either similar or can be manipulated to be nearly identical to the ones produced by the deterministic SVD, then the dependence of coordinates in the respective POD subspace on the parameters will also be similar and the resulting predictive ROMs will be competitive as measured by the predictive accuracy.*

Details of the Experiment. This experiment involves training and comparing POD and interpolation-based ROMs using POD bases produced by both deterministic (i.e., the

SVD) and randomized dimension reduction algorithms (i.e., the rSVD and the sketching-based rSVD). The experiment begins with the generation and collection of snapshots for an appropriate DoE in the parameter space. Once the full order model is evaluated for all the points in the DoE, a database is obtained in the form of the matrix of snapshots containing the field outputs to be emulated. Using a prescribed RIC (99.9% in our case), the deterministic SVD is invoked to ascertain the truncation rank k. The experiment then proceeds to split the entire dataset of snapshots into a training and a out-of-sample test set. The test set contains a small number of snapshots that are left out of all the training trials. Once ROMs with all the three dimensionality reduction variants are constructed, their accuracy is measured over this test set. Of the remaining set of snapshots, $85 - 90\%$ are chosen randomly to train the ROMs repeatedly in order to make sure that the performance of the ROMs is consistent and invariant on average to the training set. Note that the same test set is used to measure predictive accuracy for the ROMs constructed using each random set of training snapshots. The distribution of predictive performance obtained from 20 trained ROMs is reported for all the test problems using two different error metrics. For each dimension reduction variant, the distribution of mean relative error (MRE) is reported, the expression for which is given by

$$\text{MRE} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \frac{\|\boldsymbol{w}_{\text{i, predicted}} - \boldsymbol{w}_{\text{i, actual}}\|_2}{\|\boldsymbol{w}_{\text{i, actual}}\|_2} \tag{58}$$

where n_{test} is the number of solutions in the test set. Furthermore, for each test point, distribution of the so-called Spatial Mean Absolute Error (SMAE) is also reported. SMAE for test point i is given by

$$\text{SMAE}_{\text{i}} = \sqrt{\frac{\|\boldsymbol{w}_{\text{i, predicted}} - \boldsymbol{w}_{\text{i, actual}}\|_2^2}{m}} \tag{59}$$

where m is the dimensionality of the field solution $\boldsymbol{w}$. Lastly, a couple of interesting candidates from the set of test points are visualized for comparing the actual, predicted, and absolute error fields between the various dimension reduction variants.

The construction of the ROMs itself involves the following procedure depending on whether the dimension reduction algorithm is deterministic or randomized:

1. For a given DoE, given truncation rank, compute the POD basis via either the deterministic or randomized SVD algorithm

2. Express the solution in the latent space by projecting the snapshots onto the POD basis and enforce physical constraints (if possible)

3. Train surrogate models in the latent space via standard supervised learning techniques

For each random training set in each trial, ROMs are trained using each dimension reduction variant. The algorithm-specific parameters for the randomized methods are set using values found in the previous experiment. Figure 28 shows the detailed setup for this experiment.

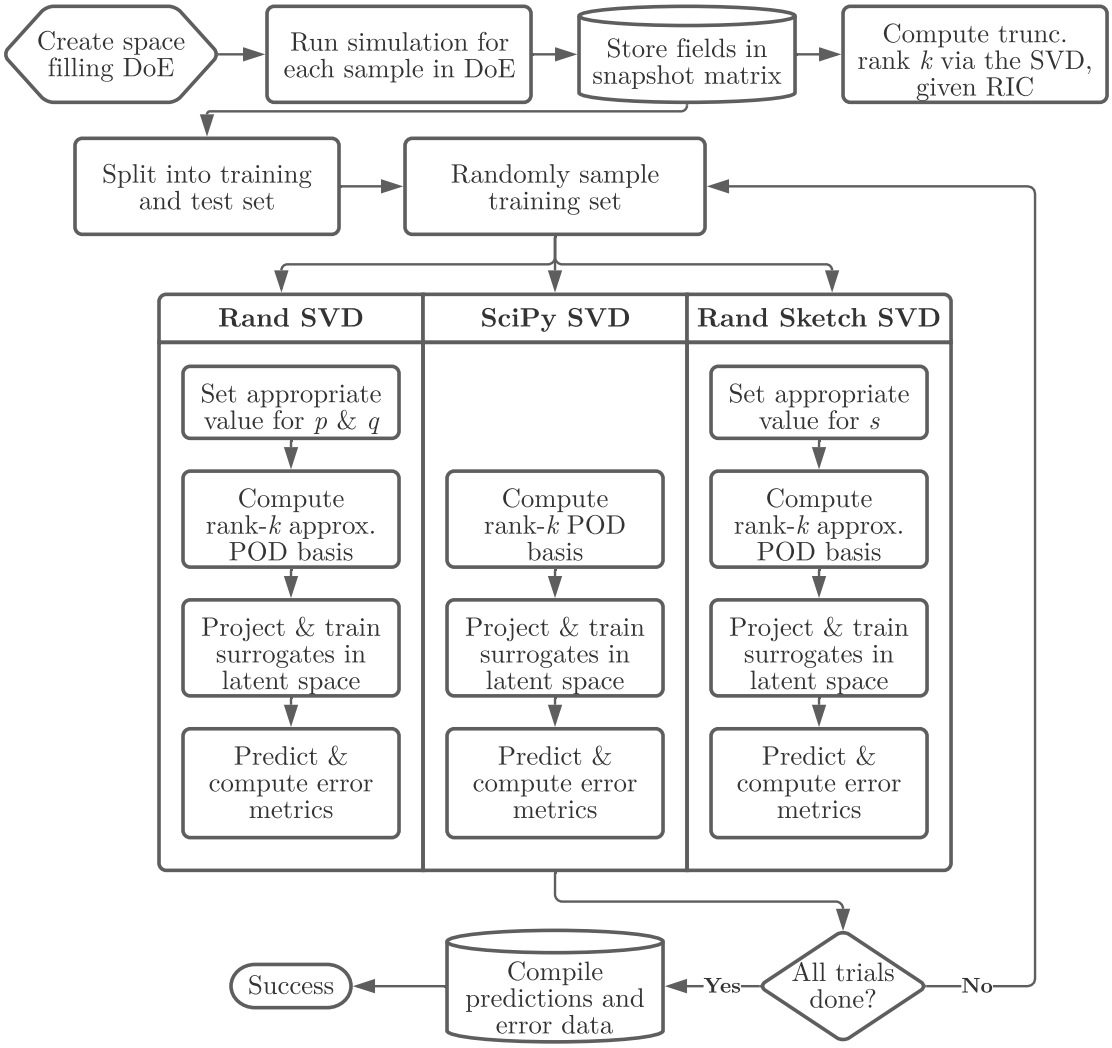


Figure 28: Setup for Experiment 1.2.

Results. Based on the results from the previous experiment, the oversampling parameter and number of power iterations for rSVD were set at 3 and 1, respectively. The sketching-based rSVD was computed using a storage factor (s) of 24. The predictive accuracy of all the three SVD variants across 20 training repetitions is shown in figure 29. It can be observed that all three methods achieve reasonably small errors even though a wide MRE spread is observed due to the disparate sets of points across the training repetitions. Although the three dimension reduction methods perform similarly on average, the deterministic SVD seems relatively more consistent in its predictive accuracy as shown by the tighter interquartile range. Both the randomized variants have a relatively wider spread in their respective errors' distributions. However, their predictive accuracy is comparable to that of the deterministic SVD method. This supports the assertion made in this experiment that if the quality of the basis obtained by randomized methods is comparable to that of the basis from deterministic methods, they can be used to create ROMs with significantly lower computational resources.

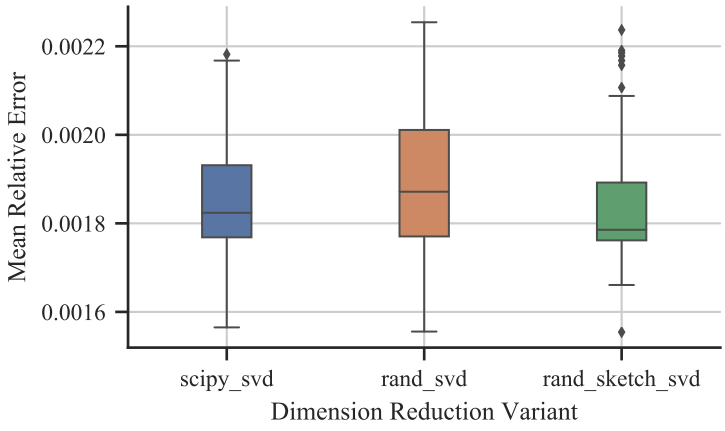


Figure 29: Mean Relative Error in the Test Set for the Canonical Problem

Figure 30 shows the SMAE across the test set for all the dimension reduction variants. It clearly shows that the majority of the predictions in the test set have small errors. For most points with a small MSAE, all the algorithms produce identical results with a relatively tight range in their MSAE distributions across the training runs. In other words, when the

prediction is accurate, the randomized methods perform similarly on average. In fact, even poor performers share a similar spread in their MSAE values. Therefore, the source of the errors may either be attributed to the latent space interpolation error and/or projection errors (because some components of solution cannot be captured by the range of the POD basis). For the visual comparison plots, test points 12 and 15 are chosen as good (fig. 31) and poor candidates (fig. 32), respectively. Visual inspection shows trivial differences in the predictions made by each of the ROMs. The patterns for the projection and regression errors are also similar between the predictions made by the different dimension reduction methods. The similarities in the absolute error distributions reinforce the fact that irrespective of the quality of the predictions, the computationally efficient randomized methods perform as well as the deterministic method.

Summary. The results for this experiment show that on average, the ROMs built using randomized POD basis perform as well as ROMs built using the deterministic POD basis. They do so while achieving a significant benefit in computational complexity and memory requirement thereby enabling construction of parametric ROMs for analyses with large field outputs under constrained computational resources. The demonstrations on the canonical problem indicate that the predictions on the test set are almost identical when the algorithm-specific parameters are set such that the quality of the randomized basis is closely comparable to the deterministic basis. In spite of setting the oversampling, power iteration, and storage factor parameters to conservative values, we observed a benefit in the time complexity of computing the SVD. Though this was done especially to support the hypothesis, it will be shown later that there is more scope for efficiency gains when the algorithm-specific parameters are set to relatively "unsafe" values, i.e., values that compromise the quality of the approximate factorization.

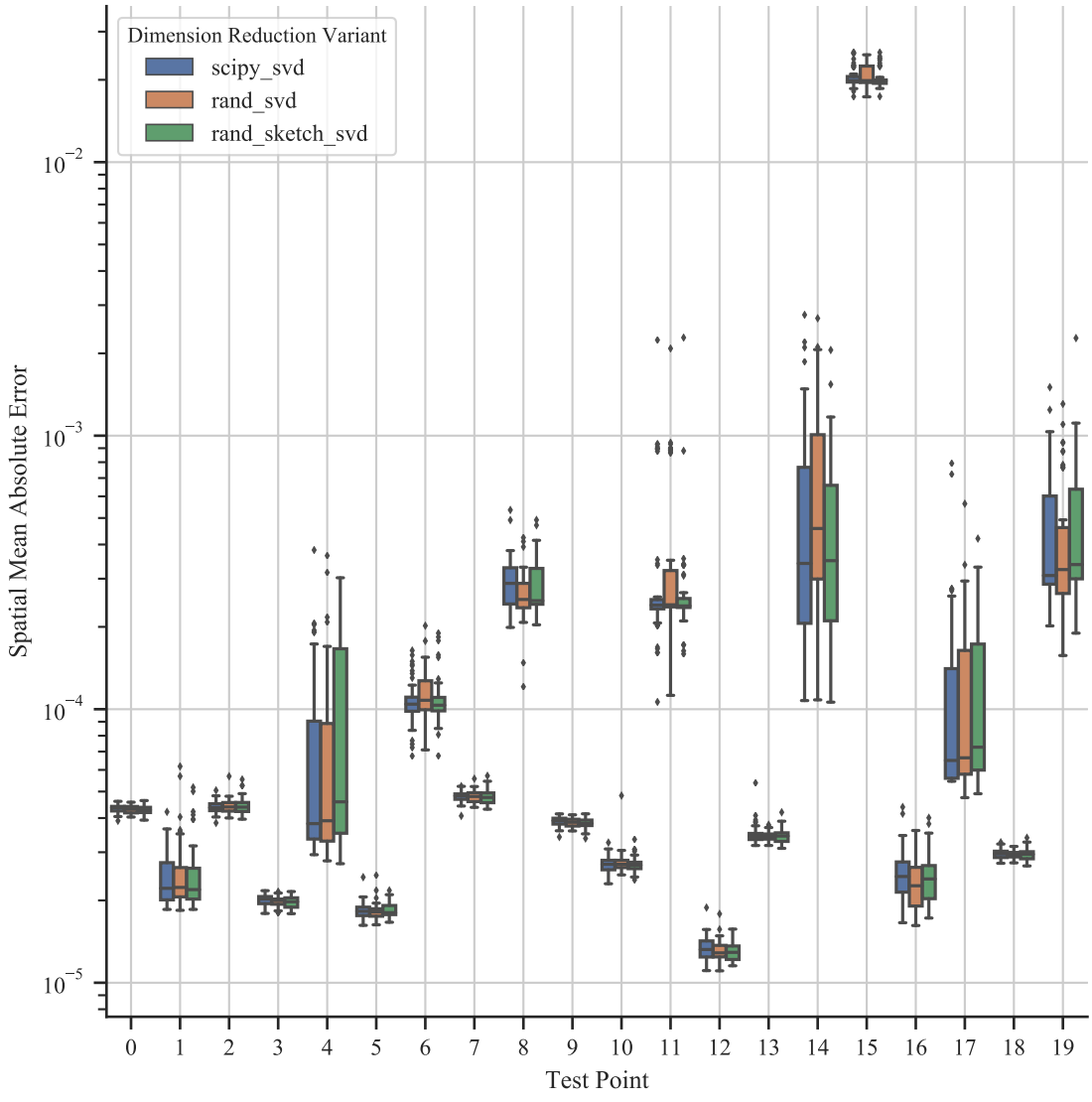


Figure 30: Spatial Mean Absolute Error in the Test Set for the Canonical Problem

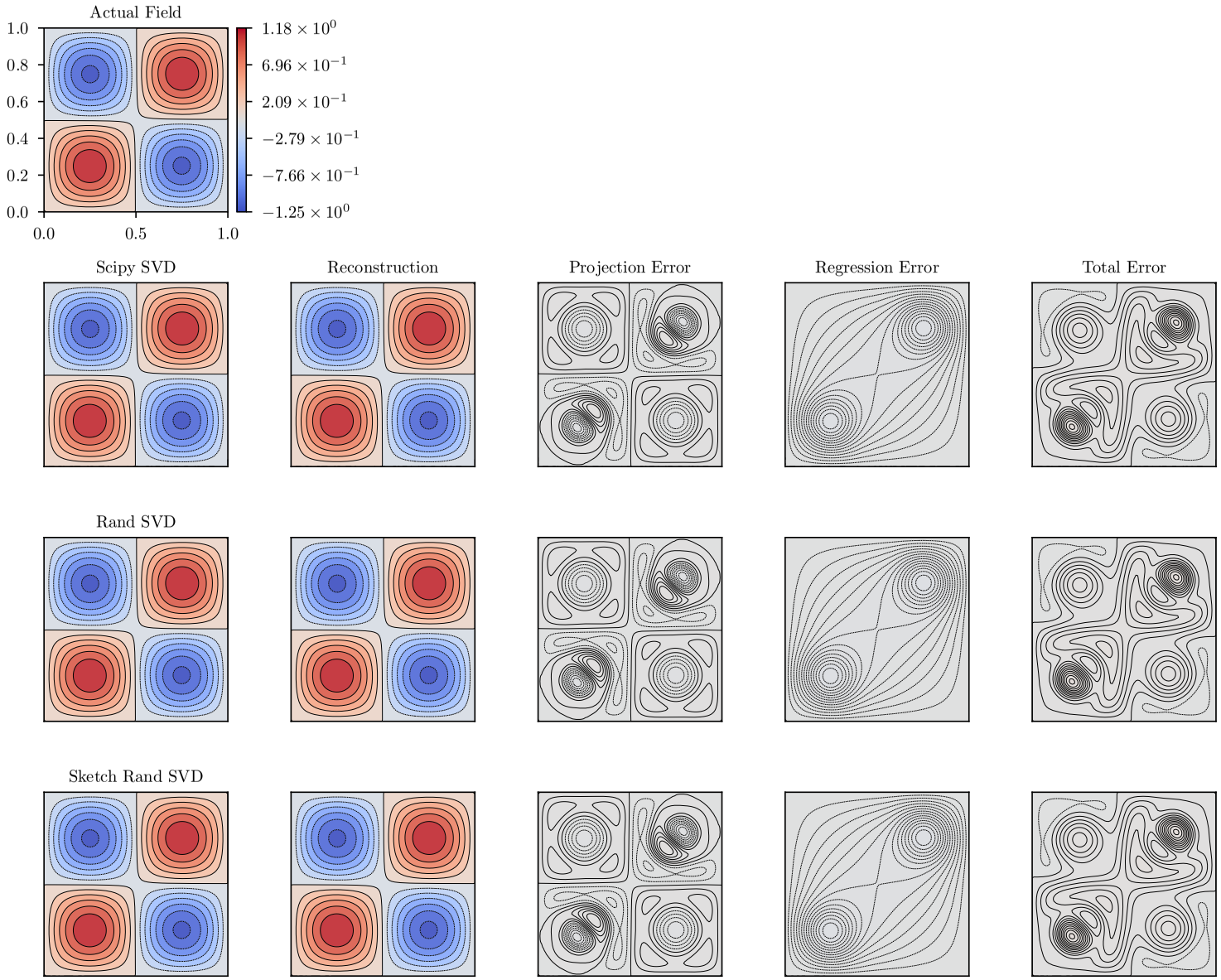


Figure 31: Visual Comparison for Good Test Point (Index 12 in Fig. 30) with $(\boldsymbol{\mu_1}, \boldsymbol{\mu_2}) = (\mathbf{3.282}, \mathbf{5.664})$)

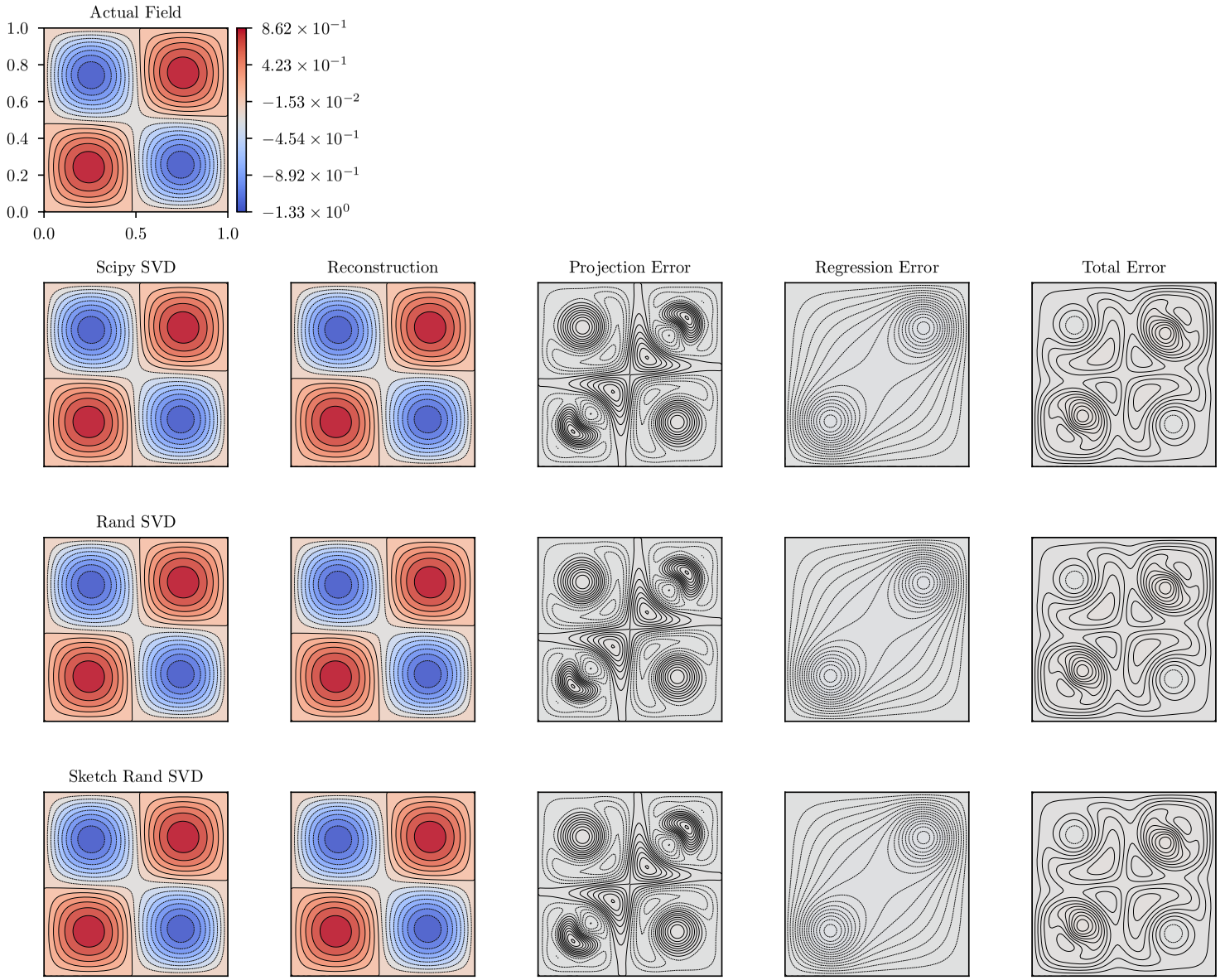


Figure 32: Visual Comparison for Poor Test Point (Index 15 in Fig. 30 with $(\boldsymbol{\mu_1}, \boldsymbol{\mu_2}) = (\mathbf{0.048}, \mathbf{6.320})$)

4.4 Application on Practical Problems

This work analyzes the performance of ROMs constructed using randomized algorithms on the flow over an airfoil and the flow over a wing. Note that any spatially distributed field quantity computed by the governing equations can be chosen for prediction purposes in order to analyze the performance of the randomized algorithms. In this work, the coefficient of pressure is chosen because it is a quantity that lends itself to intuitive visual comprehension and understanding. The presentation and analysis of results for the application problems using the fixed-rank algorithms will follow the structure presented above for the canonical problem.

4.4.1 Flow Around Airfoil

The first problem of practical interest is the aerodynamic analysis of compressible flow around the transonic RAE 2822 airfoil [77]. The flow field solution is computed using the open-source SU2 code's [84] implementation of the Reynolds-Averaged Navier-Stokes (RANS) equations and the Spalart-Allmaras (SA) turbulence model [241]. The steady-state solution is obtained using an Euler implicit scheme. As shown in Fig. 33, a structured O-grid topology is chosen to discretize the fluid domain around the airfoil.

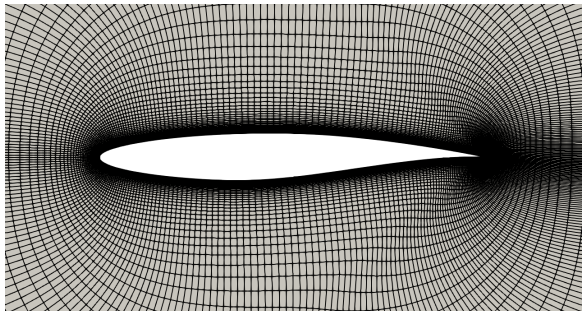


Figure 33: Baseline O-grid

The coefficient of pressure with dimensionality $m = 41796$ is chosen as the field variable to be emulated using ROMs. This test case consists of two kinds of input parameters. Two input parameters control the camber of the airfoil, which is parametrized using Free Form Deformation (FFD). The geometry is deformed by displacing the boundary nodes of an enclosing rectangular box; also called the FFD volume. The camber of the airfoil is

varied by vertically displacing the top and bottom FFD nodes at a given location along the chord together while holding the extremities still (see Fig. 34). The SU2 code provides the functionality to generate new shapes and update the grid appropriately. The final input parameter is the angle of attack which only affects the boundary conditions of the simulation. In total, this example has three input parameters $\boldsymbol{\mu} : [d_1, d_2, \text{AoA}]$ where d_1 and d_2 are the displacements of the FFD nodes from the baseline airfoil at two distinct stations along the chord of the airfoil and AoA is the angle of attack. The camber variables are limited to a maximum variation of 4% in both the positive and negative directions. The AoA is allowed to vary between 0 and 4 degrees. All the experiments with this test case use an LHS design of size $n = 1047$.

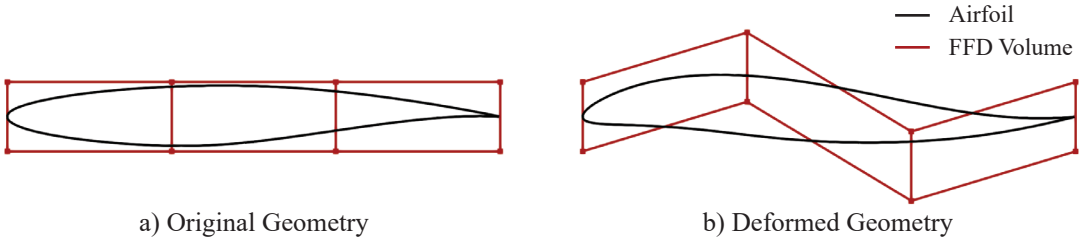


Figure 34: Camber Parametrization using FFD

Results. As shown in figure 35, the airfoil flow problem exhibits a strong low-rank structure but it is relatively weaker than the canonical problem. It is therefore amenable for creating POD-based ROMs with a significantly lower dimensionality than that of its full state. An RIC of 99.9% (typical choice in the context of ROMs) results in a truncation rank $k = 35$. The dimension reduction algorithms will compare the basis sets from a rank-35 approximation of the snapshot matrix.

Identical to the canonical problem, results for the sensitivity of the randomized SVD approximations (see fig. 36) show similar but relatively more pronounced trends in the quality of the basis obtained, perhaps because of the relatively slower singular value spectrum decay. Even for the airfoil problem, power iterations have a more significant impact on the relative reconstruction error. In addition, improvements are marginal beyond an oversampling equal to the approximation rank implying that the reconstruction quality approaches that of the

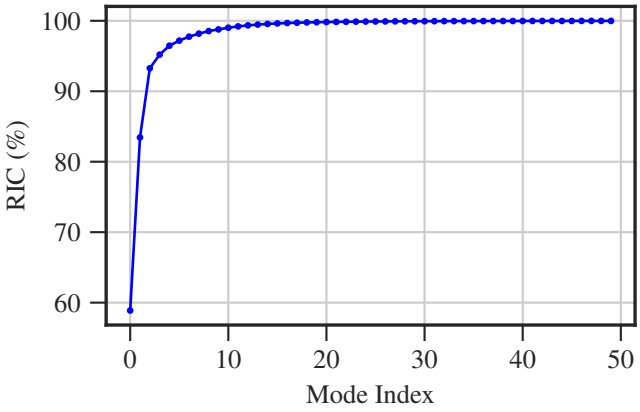
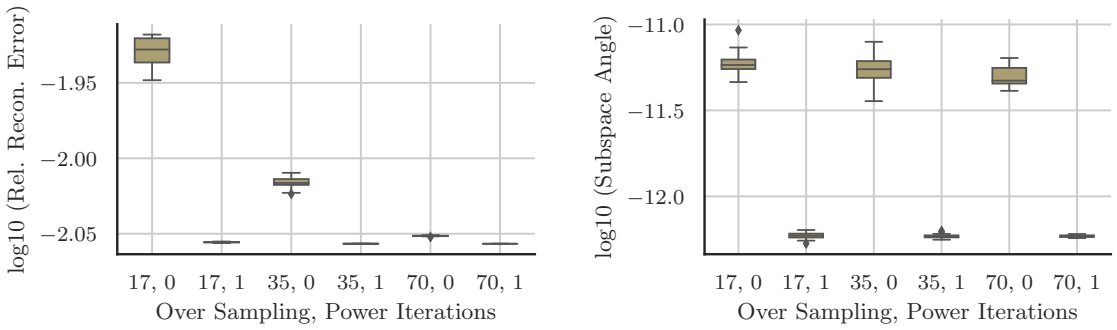


Figure 35: RIC for the Airfoil Problem

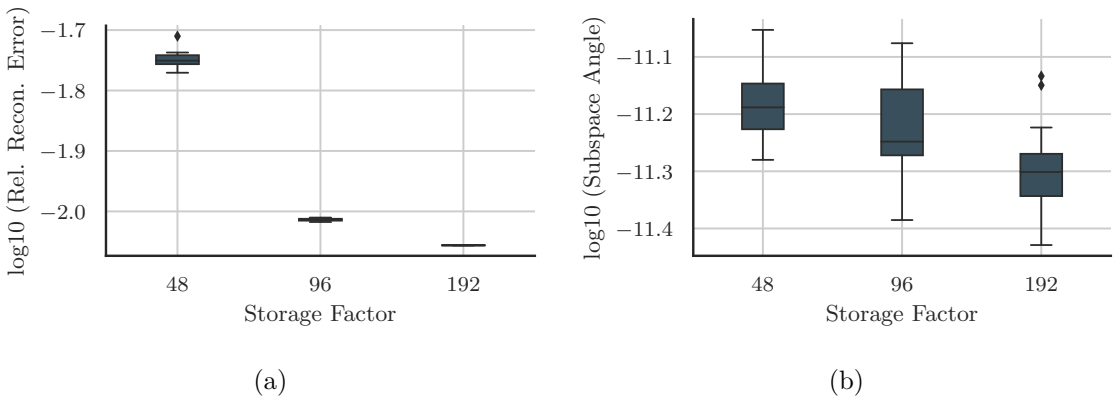
deterministic SVD. The subspace is being captured accurately with principal angles close to zero. For creating ROMs with the randomized SVD, the oversampling parameter and power iteration parameter are set to 17 and 1 respectively.

Figure 37 clearly shows that, as expected, the accuracy of the approximation increases as the storage capacity is increased for the sketching-based approximation. Beyond a storage factor of 96, there is an insignificant reduction in the reconstruction error. Note that the distributions also shrink because higher storage implies a higher probability of achieving oracle accuracy. Although principal subspace angles also clearly decrease on average as the storage budget increases, the discrepancy in the subspaces between the deterministic SVD and the sketching-based SVD is near zero across the range of storage factors. The ROMs created using the sketching-based approach use a storage factor of 96.



(a) (b)

Figure 36: Randomized SVD: Sensitivity of Projection Error and Quality of Basis for the Airfoil Problem



(a) (b)

Figure 37: Sketching-Based Randomized SVD: Sensitivity of Projection Error and Quality of Basis for the Airfoil Problem

Figure 38 shows that the MRE of the test set is similar on average. It is worth repeating that even for this problem, the sketching-based algorithm has a larger spread perhaps due to the additional compression it offers compared to the randomized SVD. As shown in figure 39, the point-wise SMAE distributions have an equal proportion of relatively poor and good predictions. However, note that for a majority of test points, the spread in the distribution of SMAEs is smaller for the deterministic algorithm than the spread in at least one of the randomized algorithms. This is in contrast to the previous observations in the case of the canonical problem. A closer inspection reveals that most of the poor performers contain shocks in the flow. Even ROMs using the deterministic SVD are known to struggle in cases with shocks and nonlinearities in the field. Therefore, even in cases where the field

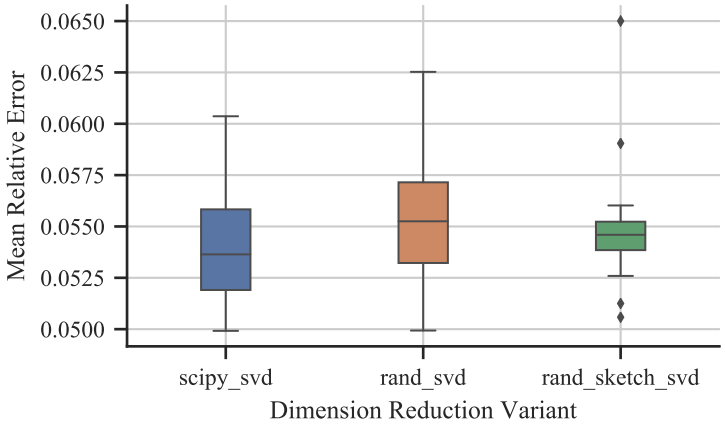


Figure 38: Mean Relative Error in the Test Set for the Airfoil Problem

is highly nonlinear, ROMs made with randomized SVD methods are comparable in quality to the ones that use deterministic SVD.

The visual comparisons in figure 41 and figure 40 for the airfoil problem are made on solutions with and without a shock respectively. Unsurprisingly, this choice automatically also differentiates them in that the solution with the shock is poor in quality. For the latter, observe that the predictions from ROMs built using all the three SVD algorithms perform equally well. In fact, the error distributions are also identical except for a few noisy patches with values close to zero for the randomized cases. In the presence of shocks (see fig. 41), all

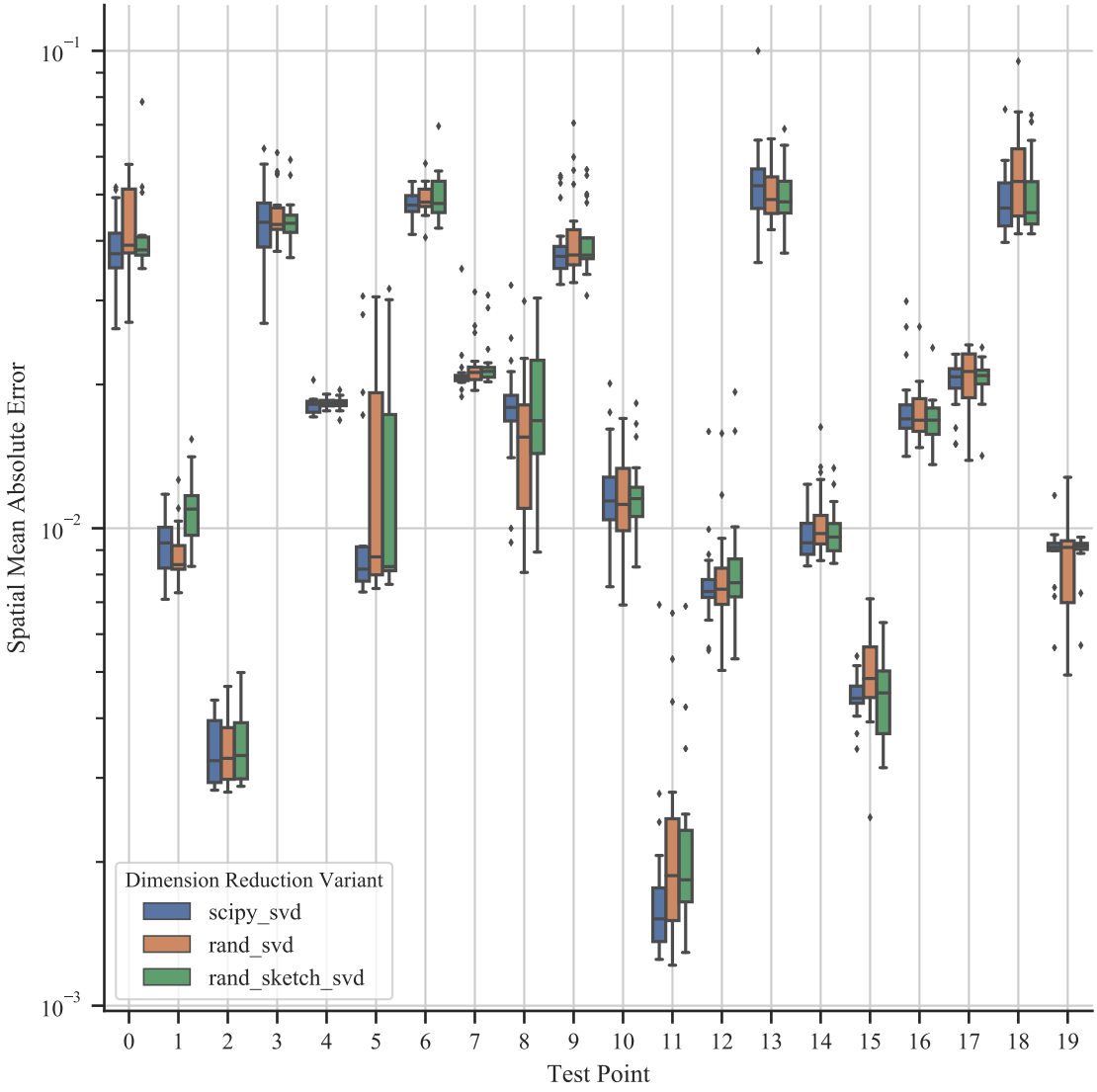


Figure 39: Spatial Mean Absolute Error in the Test Set for the Airfoil Problem

the methods struggle to accurately capture the features of the shock. However, the overall reconstruction seems to be in good agreement with the actual solution.

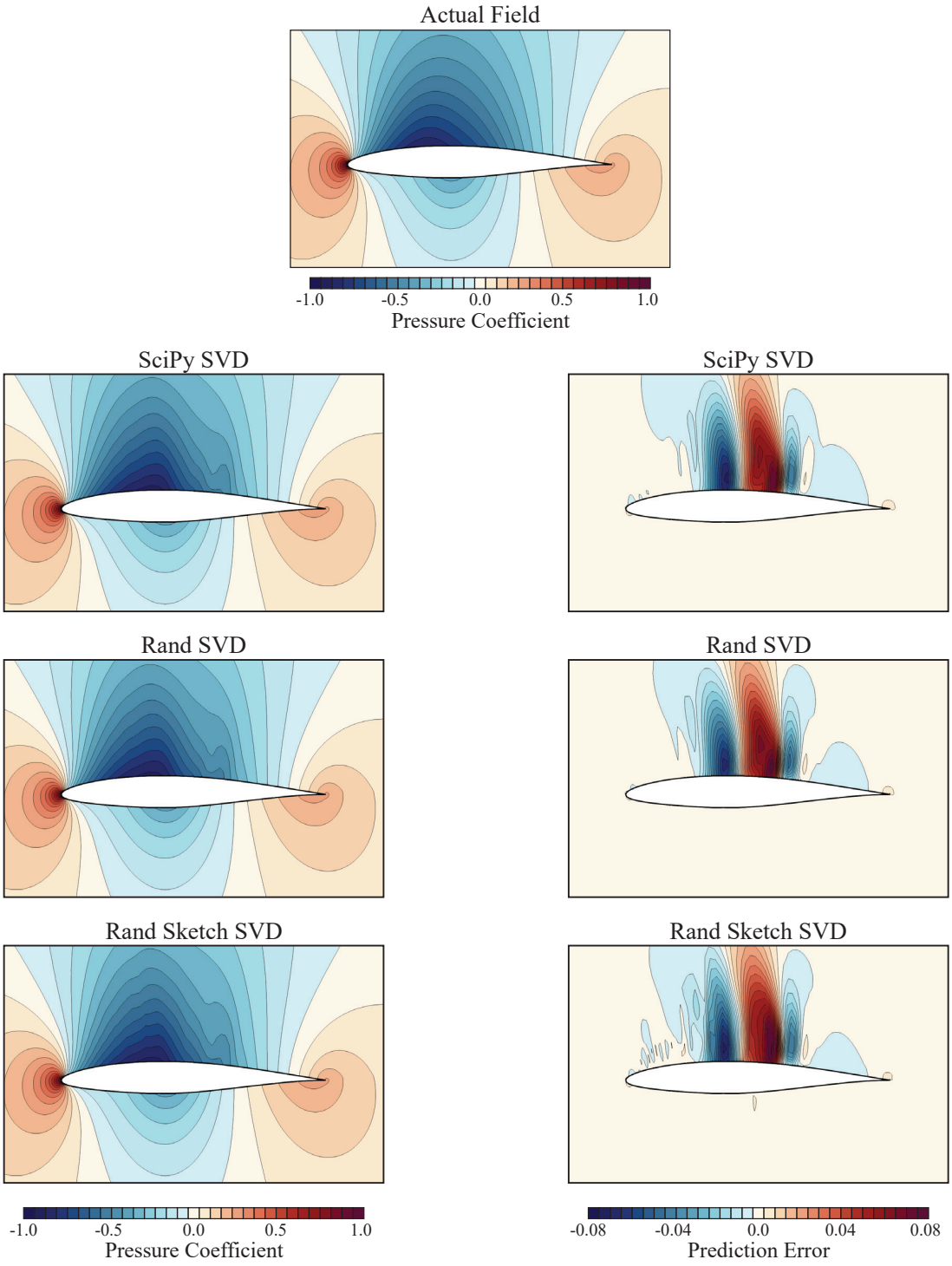


Figure 40: Visual Comparison for Good Test Point w/o Shock

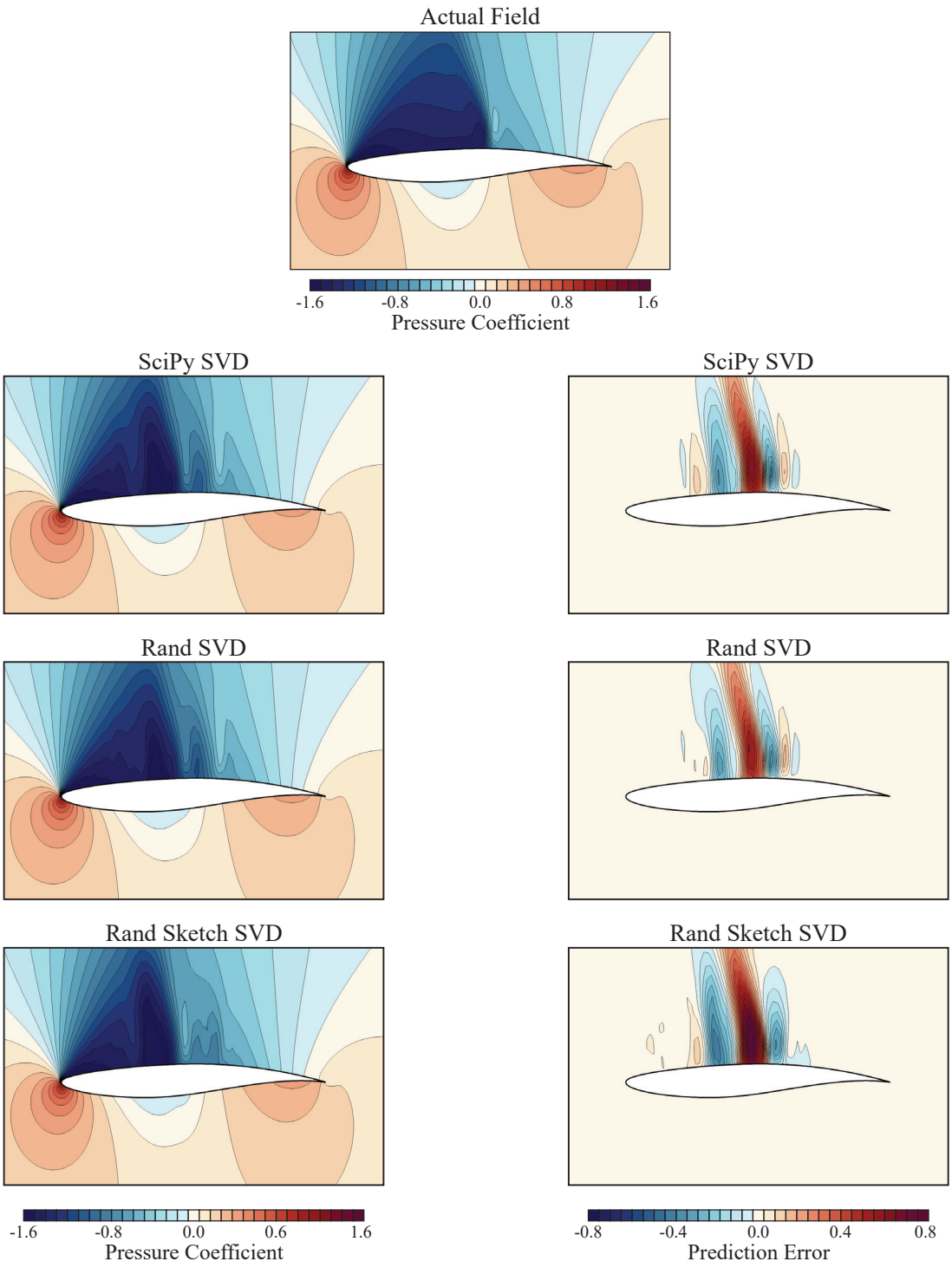


Figure 41: Visual Comparison for Poor Test Point w/ Shock

4.4.2 Flow Around a Transonic Wing

The demonstration on the second application problem involves the construction of ROMs for predicting the coefficient of pressure around the Common Research Model (CRM) wing shown in Figure 42. It differs from the previous problem in that the flow being emulated varies in three spatial dimensions. In general, such flows are relatively challenging physically and computationally. The ability to construct ROMs for such large-scale, realistic problems under limited resources truly illustrates the importance and functionality of randomization as a strategy.

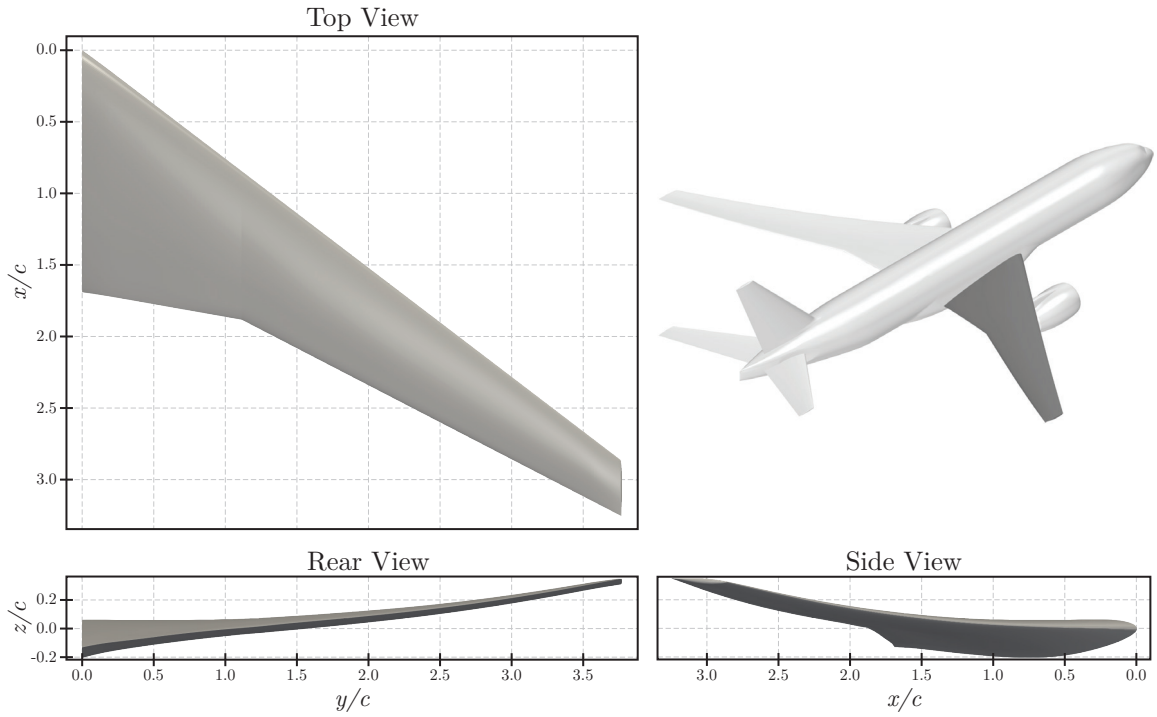


Figure 42: The CRM Wing Geometry as defined by the ADODG. The dimensions are expressed as fractions of the mean aerodynamic chord with a value of 275.8 in. The isometric view shows the wing as a part of the whole aircraft. Reproduced with permission from [194].

The CRM wing geometry used in this dissertation was initially a part of the benchmark problems developed by the AIAA Aerodynamic Design Optimization Discussion Group (ADODG) [1]. For this problem, the ROMs are constructed to emulate the coefficient of pressure at a Mach number of 0.85 and an AoA of 2 degrees. Note that the primary aim for

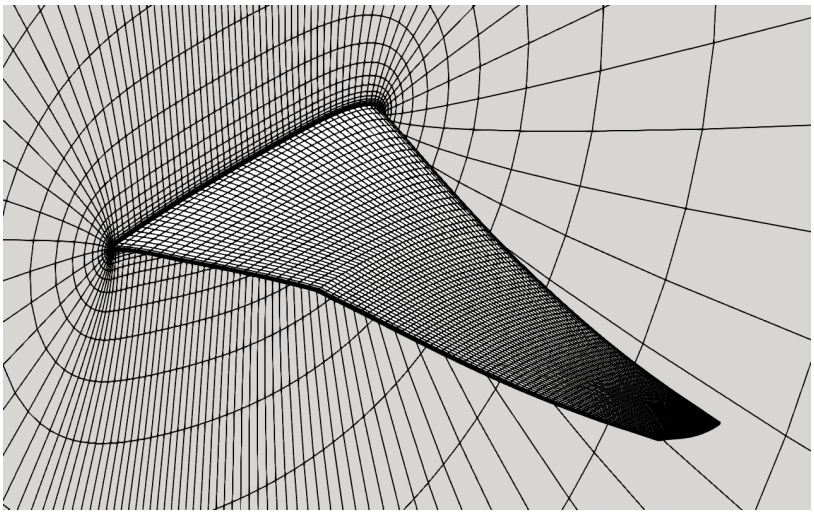


Figure 43: Grid for the CRM Wing

this research area is the demonstration of randomized algorithms to process large datasets efficiently. Therefore, the practical problem is chosen to have only a modest number of inputs. This dissertation will later present examples that have several design parameters. Two input parameters modify the shape of the wing by twisting the root and the tip by an angle of approximately ±2 deg. and ±5 deg., respectively. The wing is parametrized using a surrounding FFD box. The SU2 code is used to modify the geometry, deform the grid, and compute the RANS solution with the SA turbulence model. This test case uses an unstructured mesh provided by the Multidisciplinary Design Optimization Laboratory [2] of the University of Michigan as part of the ADODG benchmark cases. This grid of 450K cells was previously used by Lyu et al. [162] for a multilevel RANS-based optimization of

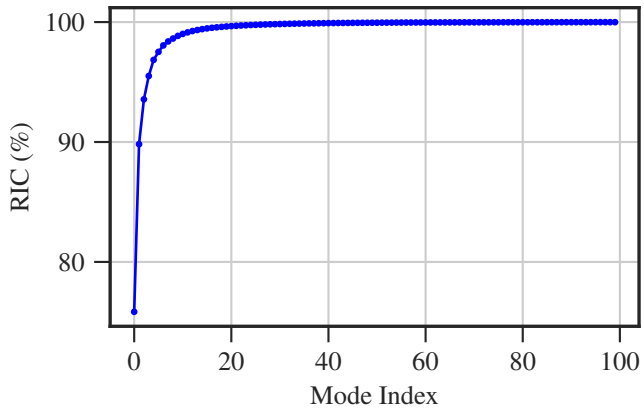


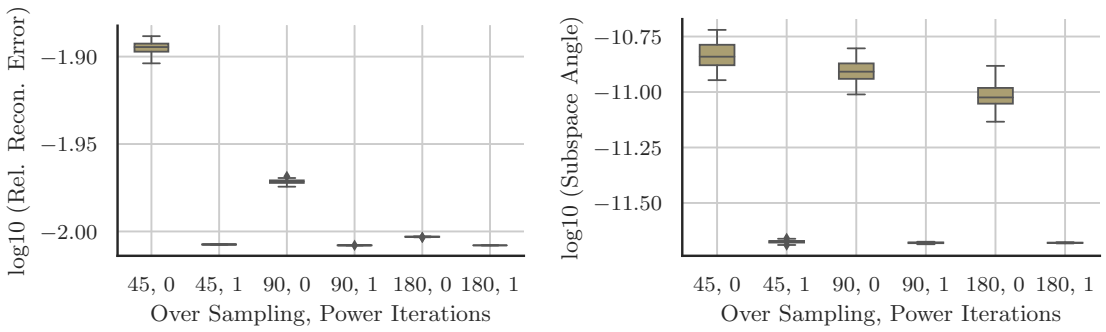
Figure 44: RIC for the CRM Wing Problem

the CRM wing. Figure 43 shows the grid used in this work. Unstructured meshes of several sizes, including the one used in this study, can be obtained from the ADODG workshop materials hosted at UMich MDO group's website. All the experiments for this problem use an LHS design of size $n = 1001$ resulting in a full snapshot matrix of size 450000×1001.

Results. Among all the problems considered, the CRM wing dataset exhibits the weakest decay in its singular value spectrum (see fig. 44). This trait is indicative of a typical large scale, practical problem which has an output large enough that it is challenging to even load into memory without hindering a personal computer's nominal performance. Additionally, for such problems, even a low rank structure usually means a relatively large truncation rank. For instance, a truncation rank of 90 achieves an RIC of 99.99% for this CRM steady-state flow problem. While a rank-90 approximation is the largest among the examples in this work, it still qualifies as low-rank in relation to the size of the CRM dataset $(450,000 \times 1001)$. The relatively slower decay in the spectrum and large dimensionality of the pressure field output make this problem the most challenging for the randomized dimension reduction methods when compared with the other problems, especially with regard to maintaining accuracy of the basis set while realizing the computational speed advantage.

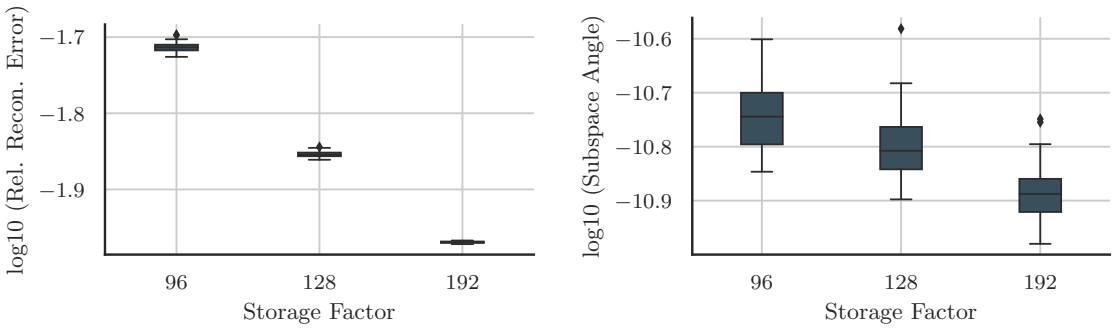
Figure 45 and figure 46 show the spread (over 50 trials) of the accuracy metrics of rank-90 subspaces computed using the randomized SVD and sketching-based randomized SVD, respectively. The performance of the algorithms is largely similar in all aspects to the airfoil problem. The advantage of using randomized algorithms is the most pronounced for the CRM wing problem because of its large size. In other words, the ability of the algorithms to efficiently produce subspaces that are comparable in accuracy to the much smaller airfoil problem is noteworthy. The oversampling and power iteration parameters are set to 90 and 1, respectively for constructing the ROM using the randomized SVD. The ROMs created using the sketching-base randomized SVD use a storage factor of 128.

The predictive performance as measured by the MRE over 50 training runs is shown in figure 47. It shows that all the three kinds of ROMs perform similarly on average. Figure 48 shows the distribution of the SMAE for the test points for the ROMs created



(a) (b)

Figure 45: Randomized SVD: Sensitivity of Projection Error and Quality of Basis for the CRM Wing Problem



(a) (b)

Figure 46: Sketching-Based Randomized SVD: Sensitivity of Projection Error and Quality of Basis for the CRM Wing Problem

with each dimension reduction method. For a majority of the test points, there is clearly no observable relationship in the distribution of SMAEs among the three dimensionality reduction variants. The minor differences seem to be driven by the randomization in the choice of the training snapshots and the minor differences in the models due to the small approximation errors introduced by the randomized dimension reduction methods. It may appear as if there is a pattern in the predictive performance (as shown by high SMAEs for the latter half of the test points in fig. 48) of the ROMs for the chosen test points. However, this apparent trend is purely coincidental and does not highlight any major problem because the behavior is identical across models built with all the dimension reduction variants. In other words, the performance (poor or good) is shared by ROMs constructed using all the dimension reduction methods across all the 50 training runs.

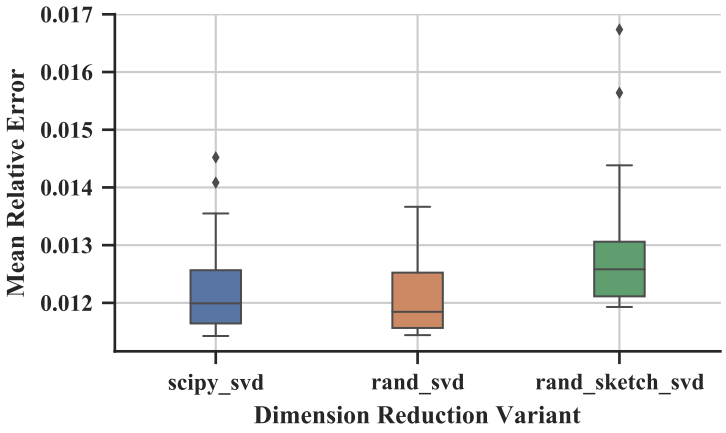


Figure 47: Mean Relative Error in the Test Set for the CRM Wing Problem

In visually comparing the coefficient of pressure on the surface of the wing for a relatively good and poor prediction test point (see fig. 50 and fig. 49, respectively), the predictive performance seems to be driven by the lack of adequate training snapshots and/or the inability of the ROM technique rather than the choice of the dimension reduction algorithm. Observe that this claim is reinforced by visually inspecting the error contours in figure 49 for the poor prediction point containing a shock. It can clearly be seen that across all the three kinds of ROMs, the location of the largest error coincides with the location of a strong shock. In general, the reconstruction agrees well with the true solution. As far as the different dimension reduction methods are concerned, the reconstructions are similar across the board even for the poor test point predictions.

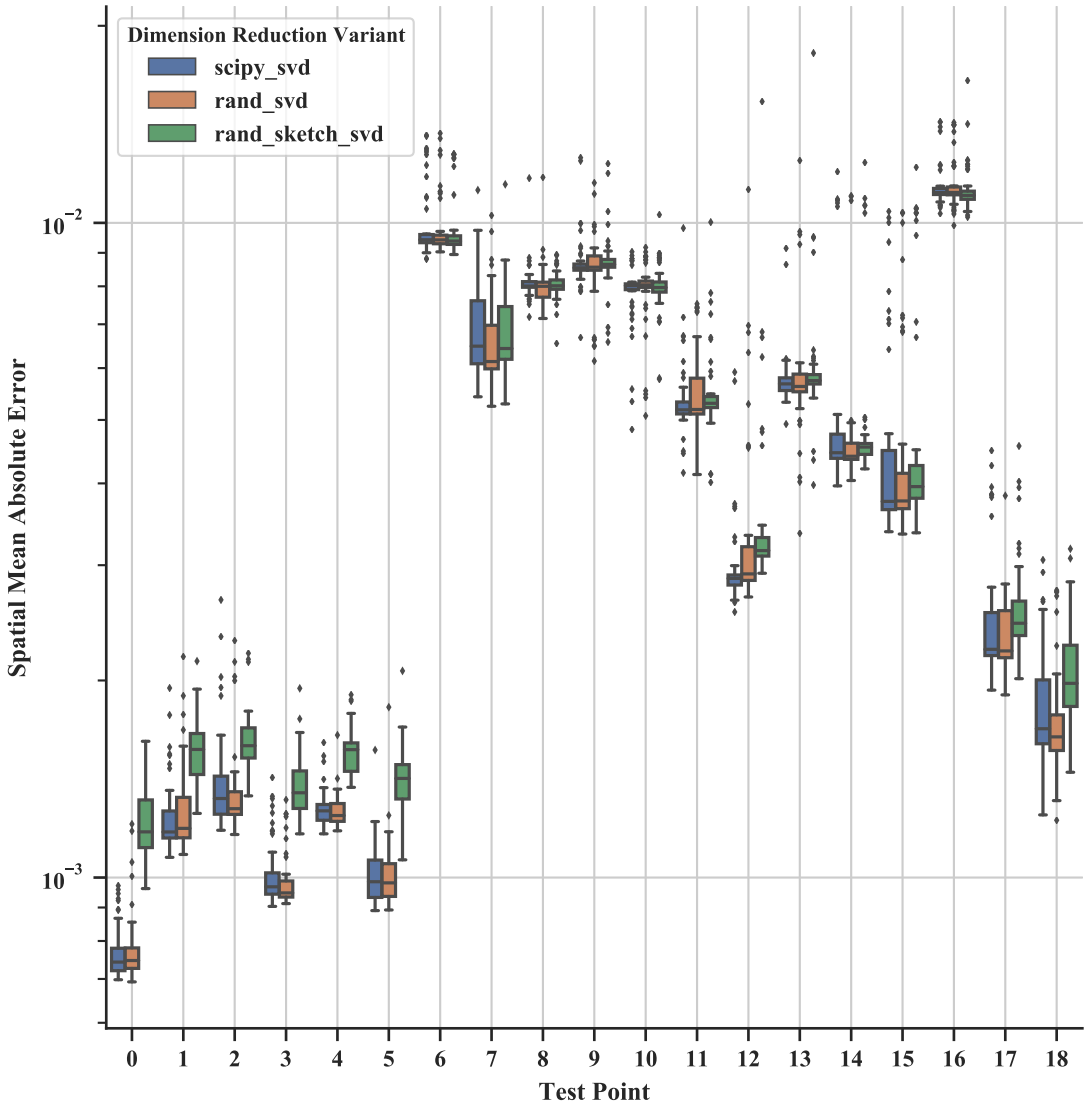


Figure 48: Spatial Mean Absolute Error in the Test Set for the CRM Wing Problem

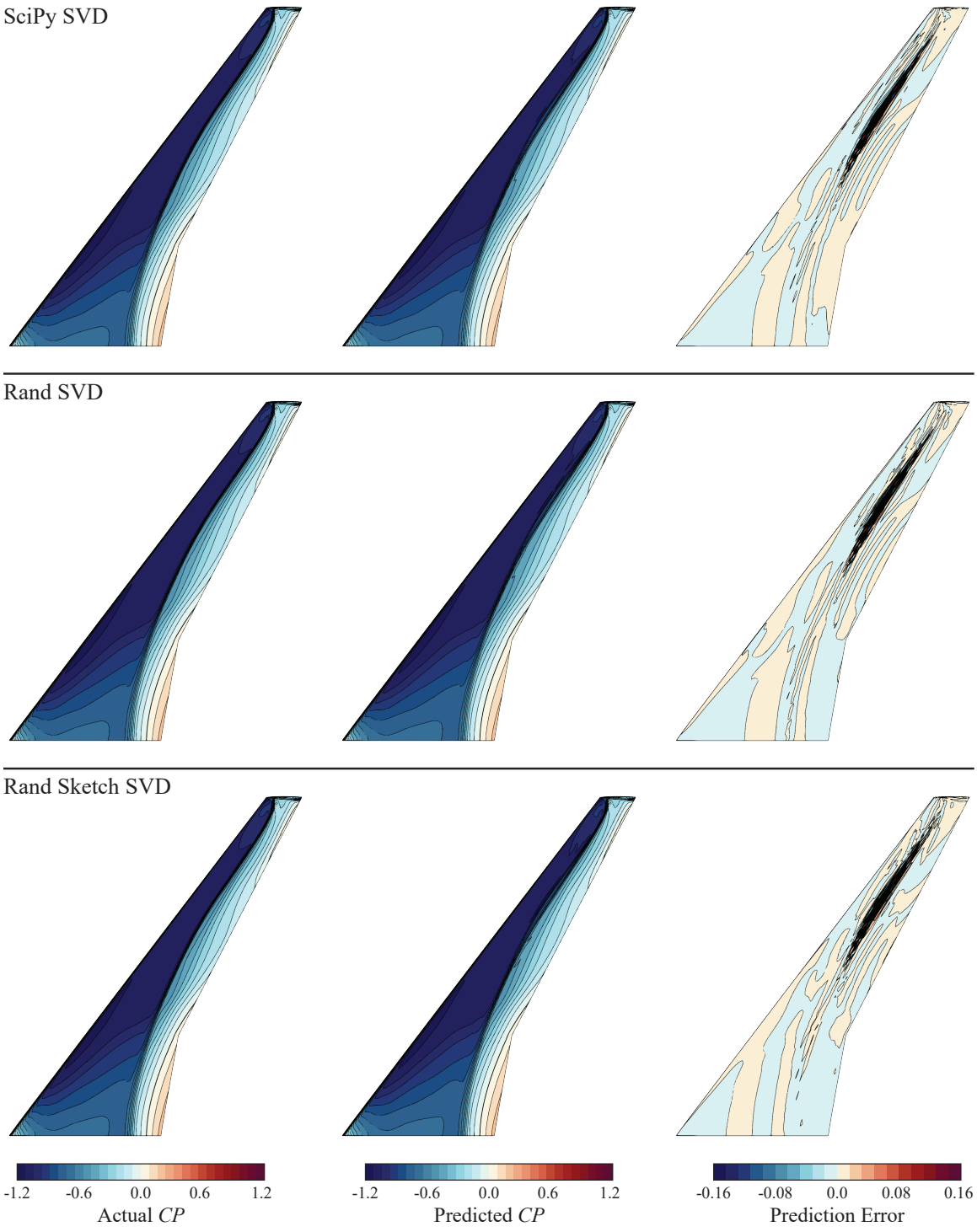


Figure 49: Visual Comparison for Bad Test Point (Index 16 in Fig. 48) with a twist of **2.28** deg. and **2.04** deg. at the root and tip respectively

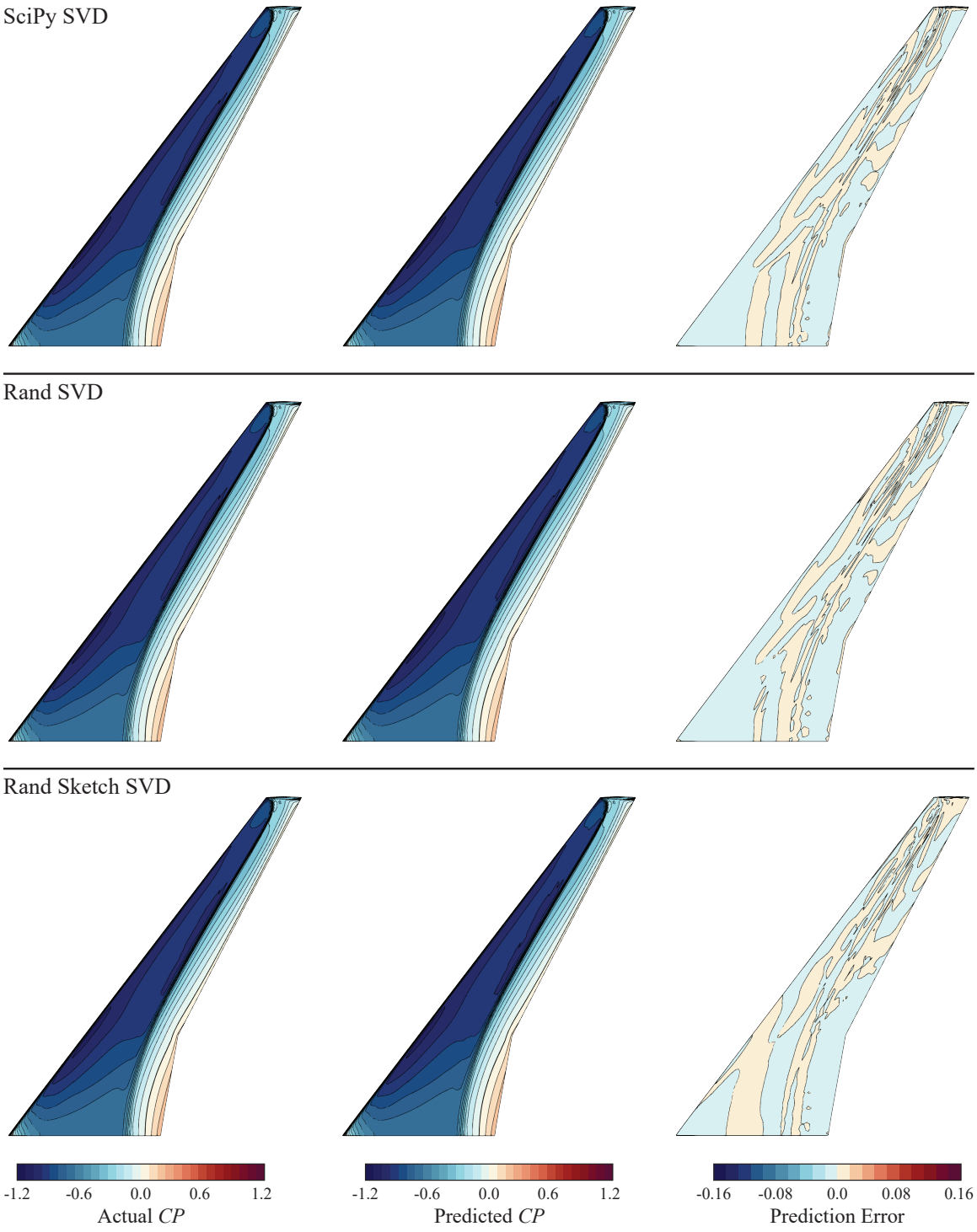


Figure 50: Visual Comparison for Good Test Point (Index 0 in Fig. 48) with a twist of **1.26** deg. and **1.09** deg. at the root and tip respectively

4.4.3 Comparison of Computational Complexity

The results presented so far show that parametric ROMs constructed using randomized dimension reduction algorithms are indeed competitive with ones built with deterministic algorithms. Once a good choice of algorithm specific parameters is made, randomized algorithms have shown to yield ROMs that have similar predictive accuracy when compared to deterministic dimension reduction-based ROMs. This section first discusses the order-of-magnitude time and space complexities of the randomized algorithms to quantify the expected gains in efficiency. Then, the wall-time for computing the POD is presented along with a summary of the algorithm specific parameters chosen for all the dimension reduction methods for each test problem considered in this work. It must be noted that the wall-time measurements are heavily dependent on the specification of the machine used to measure performance and the implementation details. Since the aim of this research is to verify whether randomized dimension reduction schemes yield good predictive ROMs, the implementation efforts did not focus on achieving optimal performance. The reported wall-times must thus be interpreted accordingly.

4.4.3.1 Space Complexity

The deterministic SVD (of a matrix A of size $(m \times n)$) algorithm requires that the entire snapshot matrix fit in-memory during the algorithm's execution, requiring $\mathcal{O}(mn)$ space. On the other hand, rank-k factorizations using randomized methods are computed by operating on significantly smaller matrices as discussed previously. Including the additional overhead to store the sampling matrices, the random sampling reduces the in-memory storage requirement to $\mathcal{O}(k \times (m+n))$ for the randomized SVD and $\mathcal{O}(k \times (m+n)+t^2)$ for the sketching-based randomized SVD. Here t is an algorithm specific parameter that is much smaller than m and n.

4.4.3.2 Time Complexity

Once the smaller matrix has been extracted from the large matrix using the sampling matrices for each of the randomized dimension reduction schemes, all the operations are

performed on this small matrix without requiring access to the original matrix. The contributions from subsequent computations and matrix multiplications to extract the small matrix result in an asymptotic time complexity of $\mathcal{O}(kmn)$ for the randomized SVD and $\mathcal{O}(kt \times (m+n))$ for the sketching-based randomized SVD. Table 4 reports the wall-time and algorithm specific parameters for the SVD computation for all the test problems as measured on a computer equipped with a 2.5 GHz Intel i7 Quad-Core processor and 16 GB of DDR3 RAM. As expected, it is evident that the wall-times for the randomized algorithms are always significantly smaller than those for the deterministic algorithm. Note that as the problem size increases, the wall-time savings are not as significant when compared with the smaller problems i.e. it seems that the wall-times do not scale well. This apparent behavior may either be attributed to the difference in algorithm specific parameter settings and/or scope for further optimization in the implementation of the randomized algorithms. Note that `scipy`'s deterministic SVD algorithm utilizes compiled libraries that are much faster than the pure python implementations for the randomized algorithms. Despite this huge difference, the reported saving in the computational time accentuates the promise of randomized algorithms.

Table 4: SVD Wall-Times and Algorithm Parameters for the Test Problems

Test Problem	Truncation Rank (k)	Mean Wall-time (s)			Algorithm Parameters		
		SVD	rSVD	sketch-rSVD	rSVD		sketch-rSVD
					p	q	s
Canonical	6	1.62×10^{-1}	1.13×10^{-2}	2.30×10^{-2}	3	1	24
Airfoil	35	4.09	4.71×10^{-1}	5.75×10^{-1}	17	1	96
CRM Wing	90	5.60×10^{1}	1.98×10^{1}	1.48×10^{1}	90	1	128

Summary. Under the assumption that the truncation rank is known a priori, the results obtained for this experiment strongly recommend the viability of randomized dimension reduction algorithms for ROMs. Our experiments saw gains in computational complexity

even with conservative estimate for the algorithm-specific parameters. However, modest values for the oversampling parameter, power iterations and storage factor were sufficient to produce accurate basis vectors that produced competitive predictive ROMs, even for practical problems with relatively weaker decays in singular values. Before concluding, one observation that distinguishes results between the canonical and practical problem must be noted. Although not of much consequence to the goal of obtaining comparable ROMs, note that the relative reconstruction errors and the predictive errors for the airfoil and CRM wing problem are an order of magnitude higher than that of the canonical problem. In general, the predictive error achieved by ROMs is problem dependent. Now that we have established the utility of randomization, let us look at another approach that replaces the requirement for specification of the truncation rank a priori with another more intuitive error tolerance parameter.

4.5 Effectiveness of Fixed-Precision Randomized Algorithms for POD and Interpolation-Based ROMs

Recall that the previous section dealt with *fixed-rank* algorithms to assess and establish the benefits of randomization. In real applications however, a priori knowledge of the exact truncation rank to achieve a prescribed RIC is impossible. A ballpark estimate is the best a subject matter expert can provide. The question that motivates the effort going forward is as follows: Is it possible to at least replace the specification of the truncation rank, if not completely dispense with it, while retain the computational gains of randomization? Stated formally, we ask the following research question as formulated in chapter 2. In essence, the question contains in itself all the relevant aspects that were investigated for *fixed-rank* algorithms.

Research Question 1.3

Can an approximate POD basis be efficiently computed by specifying the RIC instead of the truncation rank? If so, what is its effect on the computational savings offered by randomization? Does this strategy produce competitive predictive ROMs?

Fixed-precision randomized algorithms provide a promising solution to be investigated. The insight for handling the fixed-precision problem (see eq. (39)) rests on the ability to inexpensively estimate how accurately matrix $\boldsymbol{Q}$ captures the range of $\boldsymbol{W}$. The intuition for approximating the actual error $\|\boldsymbol{W} - \boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{W}\|$ is in fact based on the idea that the product of a matrix with random vectors can capture the range of the matrix. Specifically, if random Gaussian vectors $\boldsymbol{\omega}_{(i)},\ 1 = 1, 2, \ldots, v$ are used to compute the quantity $\left\|(\boldsymbol{I} - \boldsymbol{Q}\boldsymbol{Q}^*)\,\boldsymbol{W}\boldsymbol{\omega}^{(i)}\right\|$ for a small number v, the following result for the bound on the error is obtained [112],

$$\|(\boldsymbol{I} - \boldsymbol{Q}\boldsymbol{Q}^*)\,\boldsymbol{W}\| \leq 10\sqrt{\frac{2}{\pi}} \max_{i=1,\ldots,r} \left\|(\boldsymbol{I} - \boldsymbol{Q}\boldsymbol{Q}^*)\,\boldsymbol{W}\boldsymbol{\omega}^{(i)}\right\| \tag{60}$$

The probability of success for achieving this bound is more than $1 - 10^{-v}$. Note that a modest v, i.e. inexpensive matrix-vector products, is sufficient to achieve the error bound with high probability.

This optimistic error bound was recognized and exploited by authors in [169] to propose a greedy stencil to adaptively build $\boldsymbol{Q}_i = [\boldsymbol{Q}_{i-1}\ \boldsymbol{q}_i]$ (range finding stage) and $\boldsymbol{B}^T = [\boldsymbol{B}_{i-1}^T\ \boldsymbol{q}_i^T]$ until $\|(\boldsymbol{W} - \boldsymbol{Q}_i\boldsymbol{B}_i)\| \leq \epsilon$, where $\boldsymbol{B}_i = \boldsymbol{Q}_i^T\boldsymbol{W}$ and $\boldsymbol{q}_i \in \text{ran}\,(\|(\boldsymbol{W} - \boldsymbol{Q}_{i-1}\boldsymbol{B}_{i-1})\|)$. Note that $\boldsymbol{B}$ is the matrix on which the cheap factorization is computed to approximate the true factorization. If $\boldsymbol{q}_i$ is chosen as the column of $\|(\boldsymbol{W} - \boldsymbol{Q}_{i-1}\boldsymbol{B}_{i-1})\|$ with the largest norm, we get the Gram-Schmidt algorithm with column-pivoting for computing the `QR` factorization. Other choices for $\boldsymbol{q}_i$ lead to computationally inefficient optimal factorizations. Martinsson et. al. [169] proposed a random linear combination of $\|(\boldsymbol{W} - \boldsymbol{Q}_{i-1}\boldsymbol{B}_{i-1})\|$ as a choice for $\boldsymbol{q}_i$; this choice is in-line with the result in equation (60).

As one can imagine, building $\boldsymbol{Q}$ and $\boldsymbol{B}$ vector-by-vector can be extremely inefficient in terms of repeated matrix-vector products. The concept of *blocking* ensures high throughput by building matrices $\boldsymbol{Q}$ and $\boldsymbol{B}$ block-by-block. In essence, blocking takes advantage of highly optimized matrix-matrix products. For more details on blocking and its theoretical properties, the reader is referred to the brilliant explanation in [168, 169].

As briefly mentioned before, notice that the fixed-precision scheme discussed so far requires storing the residual matrix and performing several passes over the input matrix $\boldsymbol{W}$. It is indeed possible to change the algorithm in such a way that 1) the residual matrix is

not stored in memory and 2) the input matrix is accessed only once at the beginning of the algorithm. Of course, this added benefit of a single pass comes at the cost of additional loss in accuracy. The idea of how the so called single-pass algorithms work is similar to the notion of how the core approximation (matrix $\boldsymbol{C}$ in sketching-based rSVD; see eq. (45)) captures information about the row and column space of input $\boldsymbol{W}$. Recall however that the sketching-based rSVD algorithm required an estimate of the rank of the matrix, i.e. it is a fixed-rank algorithm. But in this work, the focus is the determination of the rank in an efficient manner. A single-pass efficient algorithm with the goal of solving the fixed-precision problem with a specific focus on large matrices was proposed by Yu et. al. [273]. Their work begins with the introduction of an error indicator in the Frobenius-norm, which essentially gets rid of the need to explicitly store the residual. Instead of $\|(\boldsymbol{W} - \boldsymbol{Q}_i\boldsymbol{B}_i)\|$, the proposal is to use

$$\|\boldsymbol{W} - \boldsymbol{Q}_i\boldsymbol{B}_i\|_F^2 = \|\boldsymbol{W}\|_F^2 - \|\boldsymbol{B}_i\|_F^2\,; \quad \text{see [273] for proof} \tag{61}$$

The fixed-precision stencil to approximate $\boldsymbol{Q}$ given a tolerance ϵ without storing the residual matrix can be written as follows.

Algorithm 4: Stencil for fixed-precision range finding

Input: Matrix $\boldsymbol{W} \in \mathbb{R}^{m\times n}$, tolerance ($\epsilon$)
Output: $\boldsymbol{Q}$ and $\boldsymbol{B}$ with $\|\boldsymbol{W} - \boldsymbol{Q}_i\boldsymbol{B}_i\|_F^2 \leq \epsilon^2$.

1 Initialize $\boldsymbol{E} = \|\boldsymbol{W}\|_F^2\,,\ \boldsymbol{Q} = [\,]\,,\ \boldsymbol{B} = [\,]$
2 i $= 0$ **while** $\boldsymbol{E} \geq \epsilon^2$ **do**
3 Generate block matrices $\boldsymbol{Q}_i$ and $\boldsymbol{B}_i$ that satisfy $[\boldsymbol{Q}, \boldsymbol{Q}_i]$ is orthonormal and $\boldsymbol{B}_i = \boldsymbol{Q}_i^T\boldsymbol{W}$
4 Set $\boldsymbol{Q} = [\boldsymbol{Q}, \boldsymbol{Q}_i]$, $\boldsymbol{B} = \begin{bmatrix}\boldsymbol{B}\\ \boldsymbol{B}_i\end{bmatrix}$
5 Update error estimate $\boldsymbol{E} -= \|\boldsymbol{B}_i\|_F^2$
6 i $+= 1$
7 Return $\boldsymbol{Q}$, $\boldsymbol{B}$

This stencil still involves several passes over the input matrix. Moreover, satisfying the condition in step 3 involves the orthogonalization of $\boldsymbol{Q}$ in every iteration. As such, the resulting fixed-precision scheme (not reported in detail here) has room for improvement. Yu et. al. [273] proposed an improved pass-efficient fixed-precision algorithm suited for large

matrices.

This work modifies their algorithm and empirically tests its viability for constructing parametric ROMs as follows. Once $\boldsymbol{B}$ and $\boldsymbol{Q}$ are obtained for a specified tolerance, one can technically specify an RIC which can be used to extract the approximate left singular vectors of $\boldsymbol{B}$ expanded in the range of $\boldsymbol{Q}$ corresponding to the $\tilde{k}$ largest singular values. Note that $\tilde{k}$ is approximate (and not equal to the actual truncation rank k in general) because it relies on the approximate SVD of the input matrix $\boldsymbol{W}$ computed efficiently using $\boldsymbol{B}$ and $\boldsymbol{Q}$. However, it stands to reason that if a high enough tolerance is provided, $k = \tilde{k}$ albeit at a potentially larger iteration count. Even if $k \neq \tilde{k}$ for some sensibly specified tolerance, the singular vectors corresponding to the majority of the leading singular values can be expected to have a small discrepancy when compared to the actual singular vectors. In either case, we posit that the predictive ROMs constructed using the fixed-precision scheme will be comparable with the ones built using the conventional SVD. The argument for the former case is obvious. In the latter case, because the leading singular vectors capture the majority of the features in the snapshot matrix, the inaccurately captured or completely ignored trailing basis vectors will have a negligible contribution to the predictive capacity of the ROM. Therefore, we hypothesize as follows:

Hypothesis 1.3: *Given a reasonable tolerance (such that the range of the input matrix is accurately captured), fixed-precision randomized algorithms will find a truncation rank that is either close or equal to the true truncation rank thereby enabling the specification of the RIC to efficiently compute the approximate basis vectors. If the basis vectors are comparable to the ones computed by deterministic algorithms, they can be used to efficiently construct competitive parametric ROMs.*

Algorithm 5 outlines the exact steps to find the approximate truncated SVD for a given tolerance, block size, and RIC. It essentially extracts all the information in the input matrix before the iterations begin and reformulates the re-orthogonalization step to make the algorithm pass-efficient. The result is an algorithm with more than one pass over $\boldsymbol{W}$ unlike the sketching-based rSVD. It is nevertheless designed to be pass-efficient, i.e.

minimize the number of passes over $\boldsymbol{W}$. This is the penalty we must pay to maintain a balance between accuracy, computational complexity, and the ability to find the rank of the matrix. In line with the previous findings, it is extremely beneficial to perform one or two power iterations for practical problems with a relatively flat singular value spectrum to drastically improve the factorization.

Algorithm 5: Pass-Efficient Fixed-precision Randomized Singular Value Decomposition

Input: Matrix $\boldsymbol{W} \in \mathbb{R}^{m \times n}$, RIC, tolerance (ϵ), block size (b), number of power iterations (q)

Output: Approx. $\boldsymbol{U} \in \mathbb{R}^{m \times \tilde{k}}$, approx. $\boldsymbol{\Sigma} \in \mathbb{R}^{\tilde{k} \times \tilde{k}}$, and approx. $\boldsymbol{V} \in \mathbb{R}^{n \times \tilde{k}}$, $\boldsymbol{Q}$ and $\boldsymbol{B}$ with $\|\boldsymbol{W} - \boldsymbol{Q}_i \boldsymbol{B}_i\|_F^2 \leq \epsilon^2$.

Initialize $\boldsymbol{E} = \|\boldsymbol{W}\|_F^2$, $\boldsymbol{Q} = [\,]$, $\boldsymbol{B} = [\,]$

Form $\boldsymbol{\Omega} \in \mathbb{R}^{n \times \tilde{l}}$ with i.i.d. standard normal entries . `// $\tilde{l}$ must be large enough`

`/* perform power iterations */`

for $i = 1, 2, \ldots, q$ **do**

- $\boldsymbol{G} = \boldsymbol{W}\boldsymbol{\Omega}$
- $\boldsymbol{\Omega} = \mathtt{QR}(\boldsymbol{W}\boldsymbol{\Omega})$
- $\boldsymbol{\Omega} = \mathtt{QR}(\boldsymbol{W}^T\boldsymbol{G})$

$\boldsymbol{G} = \boldsymbol{W}\boldsymbol{\Omega}$

$\boldsymbol{H} = \boldsymbol{W}^T\boldsymbol{G}$

for *i = 1, 2, ...* **do**

- $\boldsymbol{\Omega}_i = \boldsymbol{\Omega}(:, (i-1)b + 1 : ib)$
- $\boldsymbol{Y}_i = \boldsymbol{G}(:, (i-1)b + 1 : ib) - \boldsymbol{Q}\,(\boldsymbol{B}\boldsymbol{\Omega}_i)$
- $[\boldsymbol{Q}_i, \boldsymbol{R}_i] = \mathtt{QR}\,(\boldsymbol{Y}_i)$
- $\left[\boldsymbol{Q}_i, \tilde{\boldsymbol{R}}_i\right] = \mathtt{QR}\left(\boldsymbol{Q}_i - \boldsymbol{Q}\left(\boldsymbol{Q}^T\boldsymbol{Q}_i\right)\right)$
- $\boldsymbol{R}_i = \boldsymbol{R}_i\boldsymbol{R}_i$
- $\boldsymbol{B}_i = \boldsymbol{R}_i^{-T}\left(\boldsymbol{H}(:, (i-1)b + 1 : ib)^T - \boldsymbol{Y}_i^T\boldsymbol{Q}\boldsymbol{B} - \boldsymbol{\Omega}_i^T\boldsymbol{B}^T\boldsymbol{B}\right)$
- $\boldsymbol{Q} = [\boldsymbol{Q}, \boldsymbol{Q}_i]$
- $\boldsymbol{B} = \begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{B}_i \end{bmatrix}$
- $\boldsymbol{E} = \boldsymbol{E} - \|\boldsymbol{B}_i\|_{\mathrm{F}}^2$
- **if** $\boldsymbol{E} \leq \epsilon^2$ **then**
  - $\boldsymbol{U}, \boldsymbol{\Sigma}, \boldsymbol{V} = \mathtt{SVD}(\boldsymbol{B}(1 : ib, :))$
  - $\boldsymbol{U} = Q(:, 1 : ib)\boldsymbol{U}$
  - Find $\tilde{k}$ using $\boldsymbol{\Sigma}$ and RIC.
  - $\boldsymbol{U} = \boldsymbol{U}(:, 1 : \tilde{k}), \boldsymbol{\Sigma} = \boldsymbol{\Sigma}(1 : \tilde{k}), \boldsymbol{V} = \boldsymbol{V}(:, 1 : \tilde{k})$
- **else**
  - continue

Return $\boldsymbol{U}, \boldsymbol{\Sigma}, \boldsymbol{V}, \boldsymbol{Q}, \boldsymbol{B}$

Some caveats regarding the use of this algorithm must be noted with caution. First, as the number of iterations increases, round-off errors start accumulating in matrix $\boldsymbol{B}$ with every additional $\boldsymbol{B}_i$. This may lead to large inaccuracies if the input matrix $\boldsymbol{W}$ is not low-rank. For the applications considered in this work however, it is expected that the snapshot matrices are low-rank. It is reasonable to assume that the iteration count will be relatively low. Consequently, the round-off errors will also not affect the accuracy significantly. Second, if the value of $\tilde{l}$ is not sufficiently large enough for a given desired error tolerance ϵ, some more information from $\boldsymbol{W}$ must be extracted in $\boldsymbol{Q}$ and $\boldsymbol{B}$. This requires regeneration of the matrix $\boldsymbol{\Omega}$. Not only does such a situation add to the compute time for regenerating the Gaussian matrix and performing power iterations once more, it also increases the number of passes over matrix $\boldsymbol{W}$. Unfortunately, we must defer to subject matter expertise with regard to the specific problem and the need to save on compute time to set a conservative value for $\tilde{l}$. Fortunately, even in the worst case, this algorithm is relatively pass-efficient.

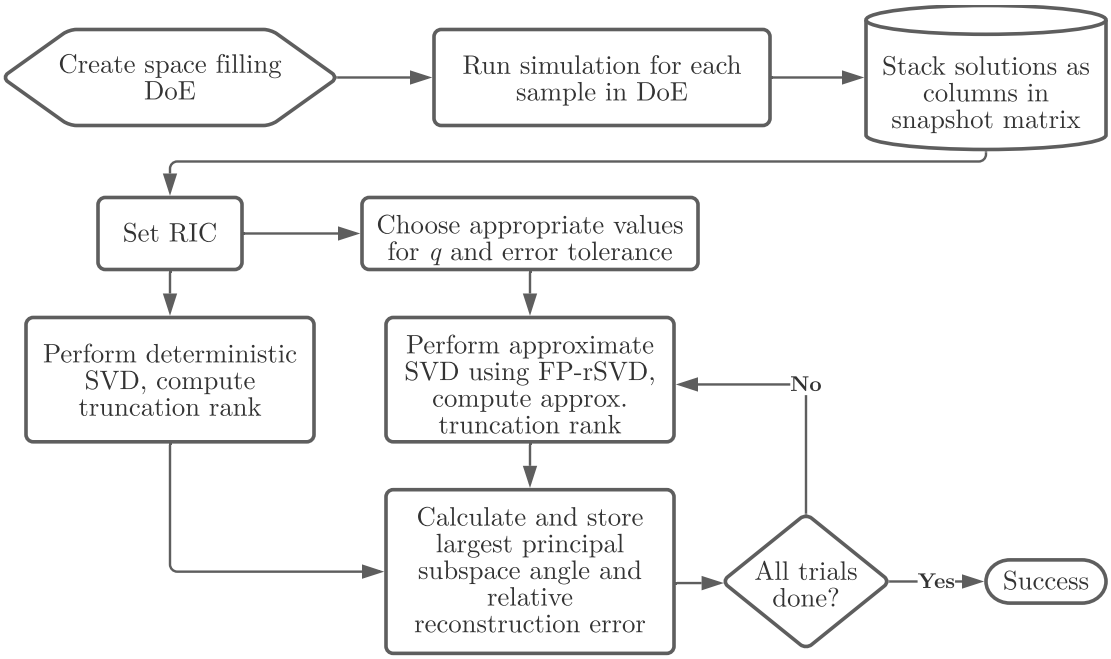


Figure 51: Experiment 1.3: Part A

Details of the Experiment. Since this experiment must combine both the analytical assessment of the basis vectors produced by the fixed-precision algorithm and the subsequent

construction and comparison of POD and interpolation-based ROMs, it is divided into two separate parts. Going forward, the fixed-precision randomized SVD will be referred to as the FP-rSVD. The first part compares the the basis vectors produced by the FP-rSVD and the second part constructs ROMs using basis vectors computed by the FP-rSVD and the deterministic SVD and compares their predictive capabilities. In the spirit of the stencil for the fixed-rank basis assessment, the first part of this experiment begins with the specification of several error tolerance levels which will be treated as the algorithm-specific parameter for the FP-rSVD algorithm. The rationale for this lies in the fact that the tolerance has a direct consequence on the computational complexity and the accuracy of the resulting approximate basis vectors. In this sense, it is analogous to the specification of algorithm-specific parameters in the fixed-rank case. Therefore, the acceptable tolerance must be varied in order to quantify the effect of algorithm-specific parameters on the trade-off between accuracy and compute time. For the chosen levels of ϵ, the FP-rSVD algorithm is invoked to compute the POD basis for a prescribed RIC. Notice that setting the RIC is analogous to the prescription of the truncation rank in the fixed-rank study. The truncation rank $\tilde{k}$ is computed using the RIC and the approximate singular value spectrum yielded by the FP-rSVD. In being able to replace the specification of the truncation rank k by the acceptable error tolerance and RIC while still saving computational time, lies the precise benefit of using the FP-rSVD algorithm. The quality and discrepancy of the obtained basis vectors are quantified using the largest principal subspace angles (eq. (52)) between the true and approximate POD subspaces and reconstruction error (eq. (53)) due to the approximate basis. One potential reason for confusion may arise regarding whether there is a difference between the relative reconstruction error (a recorded metric of quality) and the error tolerance (an input to rSVD). Recall that tolerance is specified with respect to the approximate range $\boldsymbol{Q}$ but the reconstruction error is computed using the SVD of $\boldsymbol{B}$ truncated using the RIC. As such, there is no reason for them to be identical. In fact, the reconstruction error will always be larger than the error tolerance. Figure 51 summarizes the steps for the first part of this experiment.

Once the relationship between the basis vectors computed by the FP-rSVD and the SVD

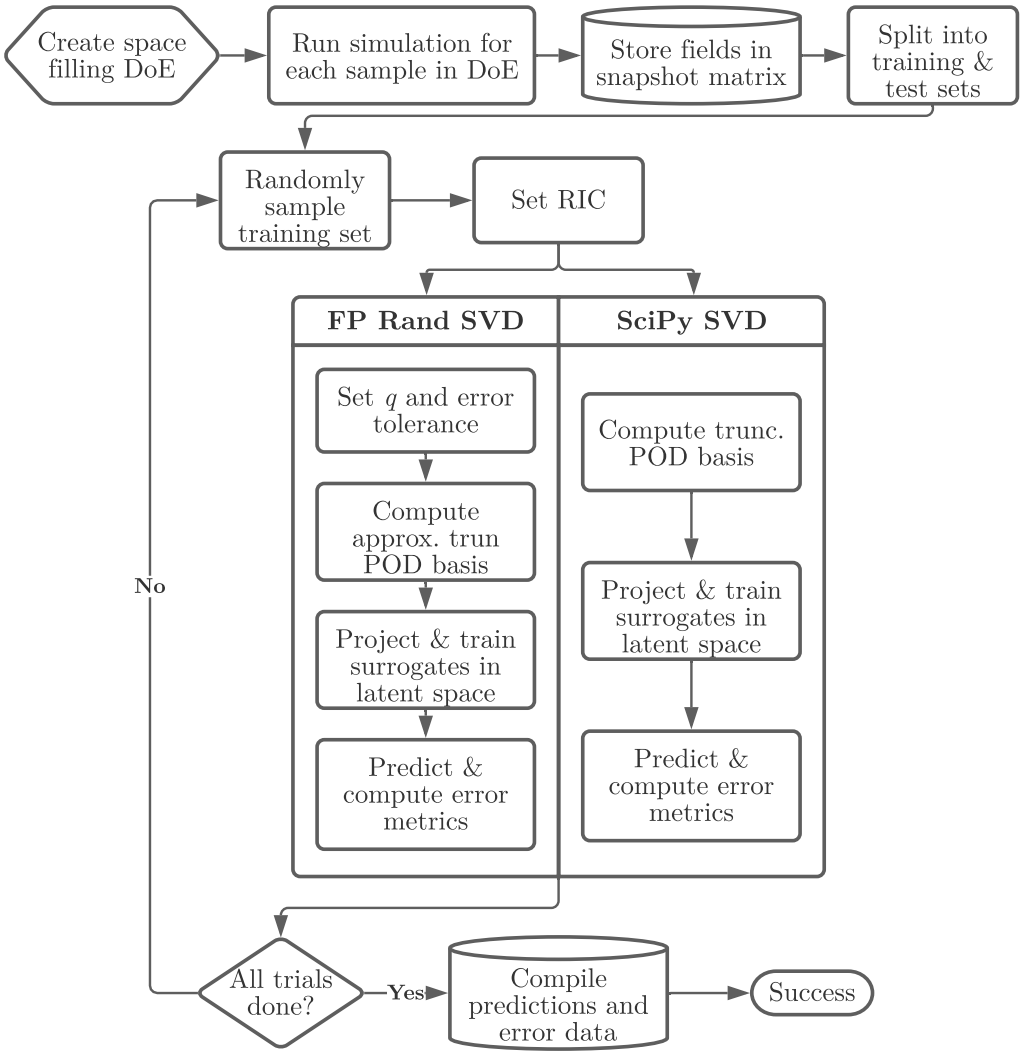


Figure 52: Experiment 1.3: Part B

is quantified, we proceed with the construction and comparison of the predictive ROMs. Similar to the fixed-rank case, we generate a space-filling DOE, evaluate the expensive model, and store the snapshots of the field variables to be emulated. For a specified RIC (say 99.9%), we invoke both the SVD and FP-rSVD to compute the POD basis, project the snapshots onto the respective POD bases and train surrogate models in the latent spaces thus obtained. Instead of specifying the truncation rank a priori for the FP-rSVD algorithm, the insights from the previous part of this experiment will be used to set values for both the error tolerance and the number of power iterations. A small number of solutions are kept out of the training process and $85 - 90\%$ of the remaining points are randomly sampled to

repeat the training process 20 times. The predictive performance of the resulting ROMs is compared using the MRE (eq. (58)) and MSAE (eq. (59)) error metrics. Figure 52 summarizes the steps of this experiment. The results for this experiment are reported on all the test problems used for the fixed-rank investigation. Let us begin with the canonical nonlinear diffusion equation problem.

Results for the Canonical Problem. Among all the randomized algorithms considered in this research area, the pass-efficient FP-rSVD algorithm is arguably the most prone to losses in accuracy because it attempts to minimize the number of passes over the input matrix and find its numerical rank. Furthermore, the algorithm tries to achieve all this while trying to save on computational time by taking a penalty in the error tolerance of the low-rank approximation. It is therefore instructive to inspect the RIC resulting from the approximate singular values computed by the FP-rSVD as shown in figure 53. Note that this visualization was not as important for the fixed-rank case because we specifically provided a rank for which a relatively high accuracy was desired. For the fixed-precision case however, the first task itself involves finding the numerical rank given an error tolerance. As such, its value and the approximate singular values computed by FP-rSVD are affected by the error tolerance and the number of power iterations. This places an upper bound on

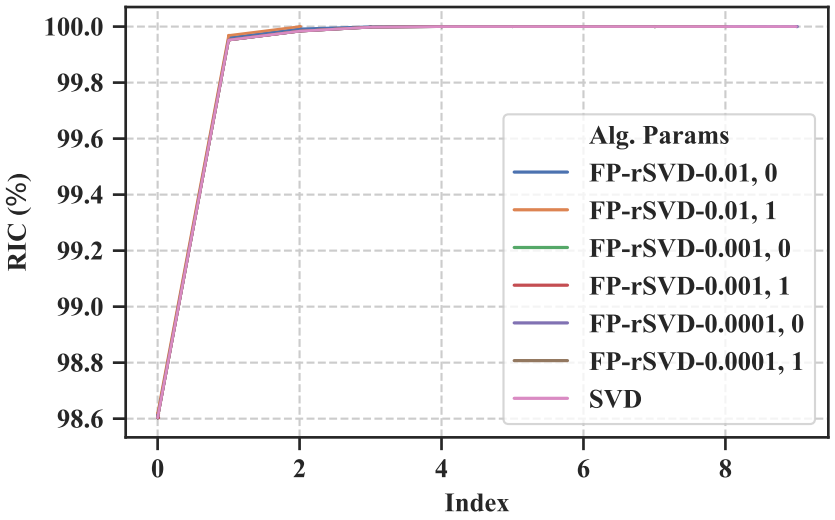
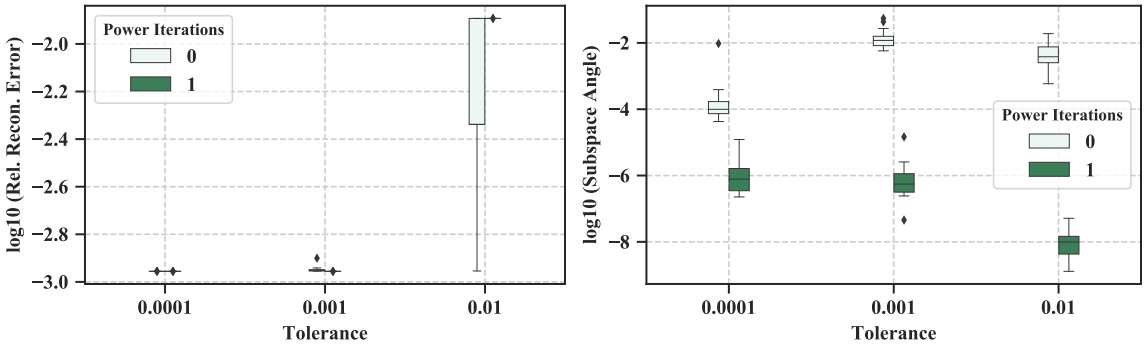


Figure 53: Comparison of RIC for the Canonical Problem

the approximate truncation rank $\tilde{k}$ and results in an RIC that is different than the actual RIC at a given index. Because the canonical problem is extremely low-rank and the first few singular vectors capture a vast majority of the information in the snapshot matrix, the inaccuracies appear insignificant across the range of acceptable error tolerances and power iterations (see fig. 53). Moreover, the already insignificant differences in the RIC only begin to appear beyond the first two modes for the low-rank approximation. In fact, they appear at relatively high values of the acceptable error tolerance.

With $\epsilon = 0.0001$ and $\epsilon = 0.001$, the FP-rSVD consistently computed a rank 5 POD basis i.e. $\tilde{k}$ was found to be 5. In fact, for the canonical problem dataset used in this experiment the truncated SVD also determined the truncation rank k as 5 for an RIC of 0.999. Based on the observations for this low-rank physical system (ideal for ROM construction), the preliminary evidence suggests that the FP-rSVD algorithm is able to accurately find the numerical rank and the truncation rank consistently.



(a) (b)

Figure 54: Fixed-Precision Randomized SVD: Sensitivity of Projection Error and Quality of Basis for the Canonical Problem

Figure 54 shows how the relative reconstruction error and the largest angle between the actual and the approximate POD subspaces vary as the acceptable error tolerance is increased. The box plots represent the results from 20 trials of the FP-rSVD algorithm. Observe that an $\epsilon = 0.001$ gives an almost identical reconstruction error. Because of the low-rank nature of the problem, the power iteration parameter does not have a significant effect on the accuracy. At $\epsilon = 0.01$, the power iterations make the result more consistent. This implies that not performing any power iteration is actually helpful for extremely low-rank

systems as observed by the interquartile range. Recall that power iterations decrease the effect small singular values have on the factorization error by repeated exponentiation. But when the system is low-rank and ϵ is large, the information in basis vectors corresponding to the seemingly small yet relatively significant singular values gets discarded. This occurs primarily because a large acceptable tolerance causes large inaccuracies in the approximate singular values resulting in a more rapid decay.

Another major difference in the performance of the FP-rSVD algorithm can be seen in the difference in the subspace angles between the fixed-rank case (see figures 26 and 27) and the fixed-precision case (see figure 54). Although the values of the largest angles themselves are near zero for all practical purposes, this behavior may pose challenges for practical problems. The effect of the power iteration is relatively more pronounced for the quality of the resulting basis than it is for improving reconstruction error. Therefore, it is an important parameter to control the quality of the basis vectors which is especially consequential for the performance of the ROMs.

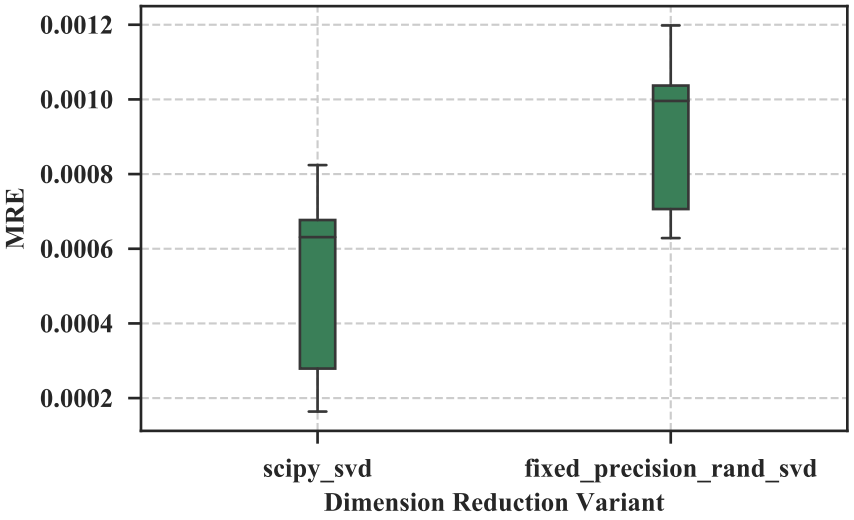


Figure 55: Mean Relative Errors in the Test Set for the Canonical Problem

Figure 55 shows the predictive performance of 20 ROMs built using both the FP-rSVD and SVD algorithms. The FP-rSVD algorithm is run with acceptable error tolerance $\epsilon = 0.001$, 1 power iteration, and a block size of 10. These specific values found the true truncation rank of 5 with high probability as measured in the 20 trials. It is clear that the

resulting ROMs have marginal differences in their predictive accuracy, the deterministic SVD algorithm being more accurate. Although the plots show that the deterministic model may possibly be twice more accurate in the worst case, the errors themselves are small. It is safe to assume that any application that hinges on the ability to distinguish between such small errors does not rely on nonintrusive ROMs. The SMAE results highlight the test set candidates that contribute to the difference, albeit minor, in the MRE. Figure 56 shows that models using the FP-rSVD for dimension reduction incur larger spatially distributed errors on average for most of the points in the test set. Only a minority of the test points

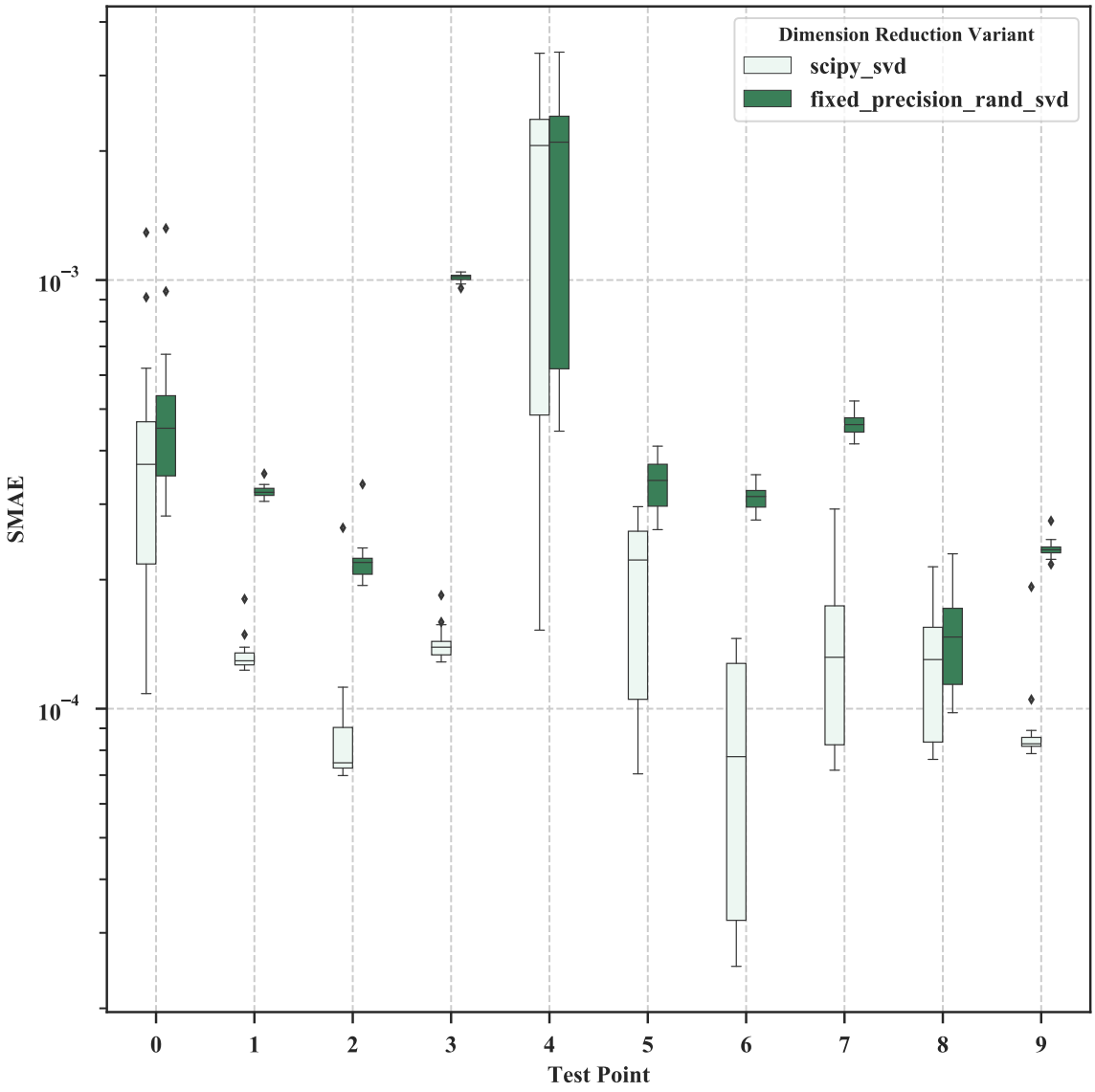


Figure 56: Spatial Mean Absolute Error in the Test Set for the Canonical Problem

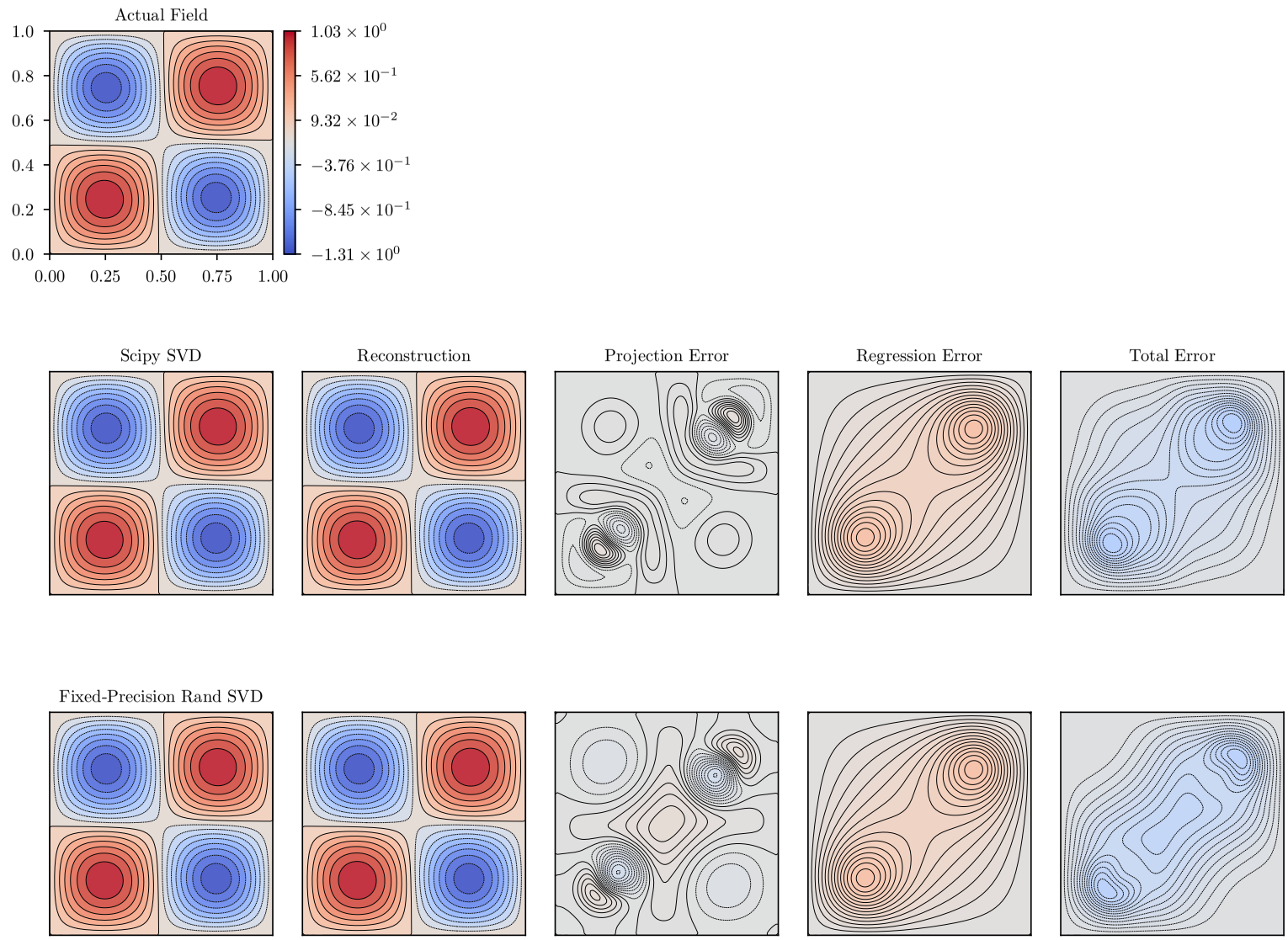


Figure 57: Visual Comparison for Test Point with the Worst Prediction Error (Index 15 in Fig. 56)

have a comparable accuracy between models built using the SVD and FP-rSVD algorithms. However, the difference is small for most of the points with the exception of test point 3 which is visualized in figure 57. A cursory glance does not reveal any difference between the actual and predicted solutions. On closer inspection of the error contours, we can see that the total error for the FP-rSVD prediction has a relatively higher mass in the center when compared with the SVD prediction. While the regression error contours appear visually identical, there is a difference in the projection error contours indicating that the basis vectors computed by the FP-rSVD introduce relatively larger errors when this particular solution is projected and reconstructed.

Results for the Airfoil Problem. For the airfoil problem, the RIC was set at 0.9999 resulting in an actual truncation of 55. Recall that the ROMs for the airfoil problem in the fixed-rank studies were built using an RIC of 0.999. The increase in the RIC is a deliberate attempt to test the effectiveness of the FP-rSVD algorithm. The airfoil problem is characterized by a rapid initial decay of singular values. The rate of decay decreases rapidly towards the later indices. Although the system is still low-rank when compared to the number of spatial unknowns and the number of solutions, this problem is more challenging than the canonical problem. Despite the added challenge, the RIC values computed by

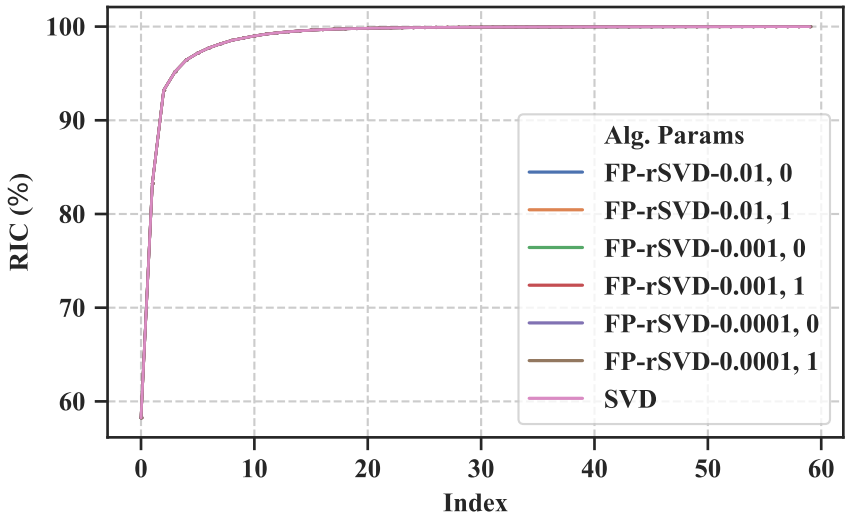
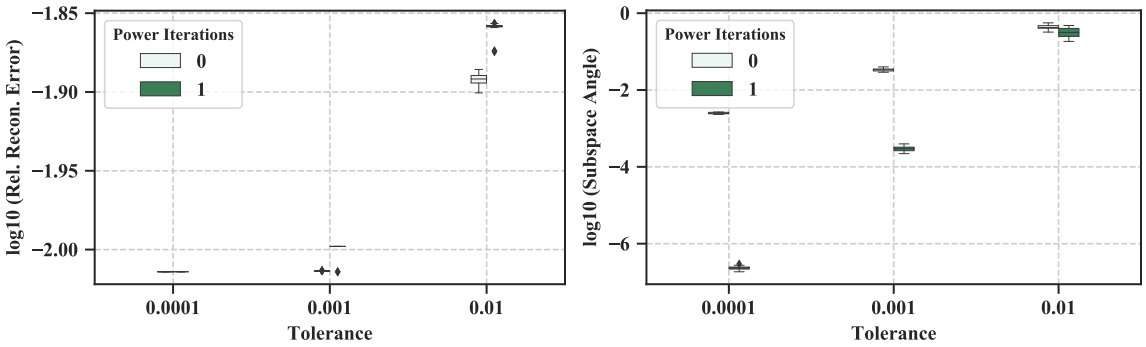


Figure 58: Comparison of RIC for the Airfoil Problem



(a) (b)

Figure 59: Fixed-Precision Randomized SVD: Sensitivity of Projection Error and Quality of Basis for the Airfoil Problem

the FP-rSVD in figure 58 do not show significant visual differences when compared to the actual RIC values. However, note that beyond an index approximately between 10 and 15, the singular values start contributing insignificantly to the RIC. As a result, errors in the relatively flat trailing portion of the spectrum can cause the approximate truncation rank $\tilde{k}$ to be different from the actual value for a given RIC. As long as a relatively large ϵ introduces small errors in the singular values beyond index 10-15, the resulting $\tilde{k}$ should be sufficient to capture the majority of the variance in the snapshot matrix assuming the POD basis vectors are of good quality. Observe in figure 59 that at higher values of ϵ, increasing the number of power iterations affects the reconstruction error adversely. This non-intuitive

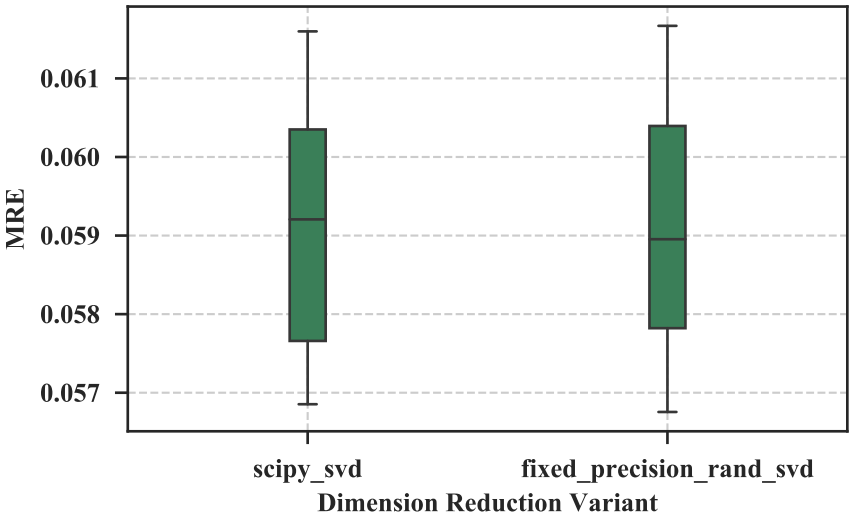


Figure 60: Mean Relative Errors in the Test Set for the Airfoil Problem

behavior is similar to what is observed in the case of the canonical problem. A large number of power iterations causes $\tilde{k}$ to be smaller for a given RIC because the trailing small singular values are ignored. In turn, a small $\tilde{k}$ contributes to the high reconstruction error. At very small values of ϵ however, power iterations have a negligible effect on the reconstruction error because the FP-rSVD strives to approximate the low-rank factors accurately. Also, note that the subspace angles for the airfoil case are also much larger than what was observed for the fixed-rank studies. In fact, at $\epsilon = 0.01$, the largest angle is far from 0. But large values of ϵ first start affecting the trailing singular values and vectors. Therefore,

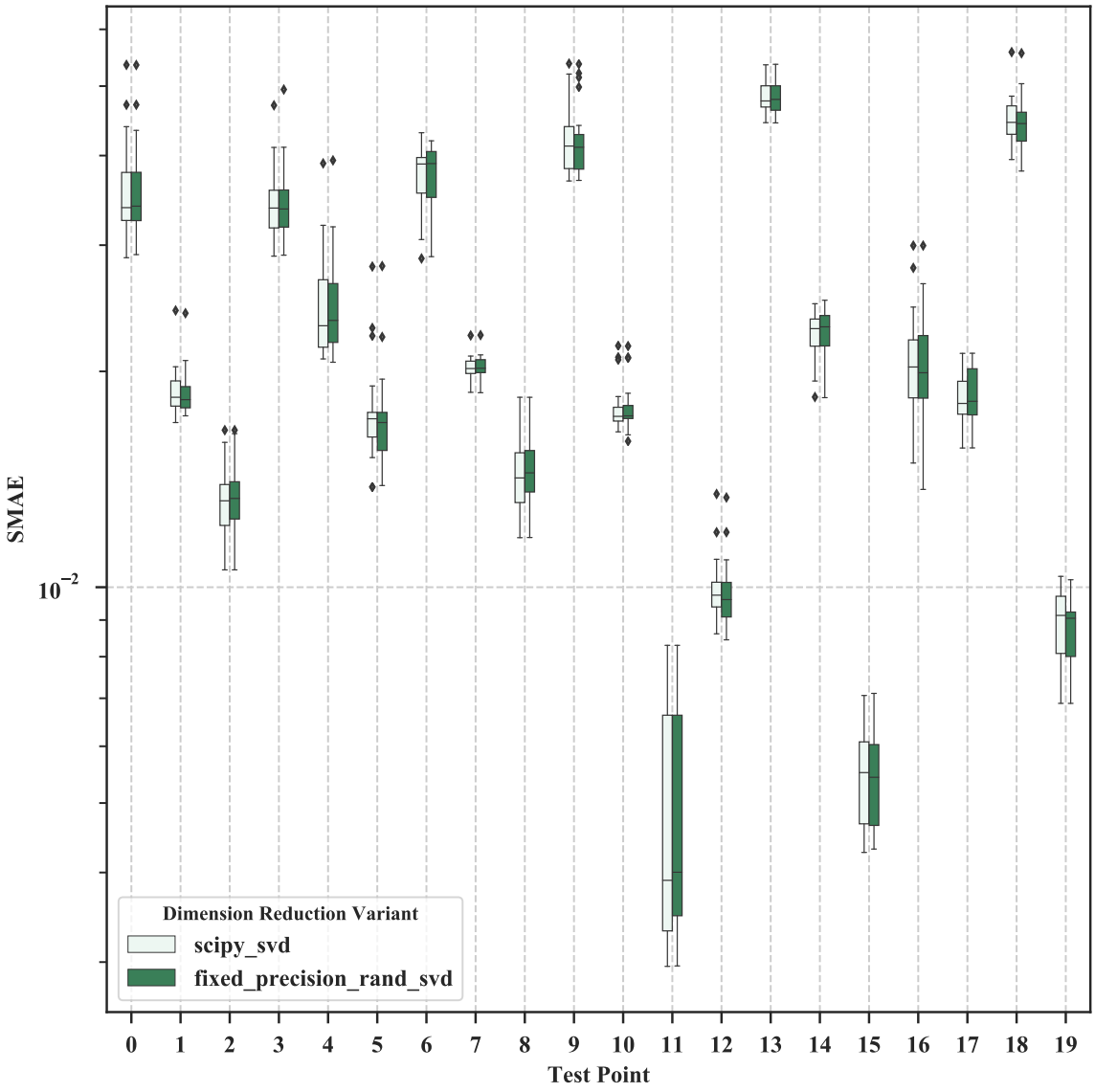


Figure 61: Spatial Mean Absolute Error in the Test Set for the Airfoil Problem

as long as the acceptable error tolerance is sufficient to capture the first 10-15 modes as measured by the subspace angle metric, the resulting ROMs should be competitive. For the airfoil problem, the values for ϵ and number of power iterations were set at 0.001 and 1 for constructing ROMs using the FP-rSVD algorithm. The block size b was set to 10. These settings consistently resulted in an approximate truncation rank $\tilde{k} = 46$. Note that this is our requirement that $\tilde{k}$ be at least greater than $10 - 15$. Results obtained over 20 random training trials are reported.

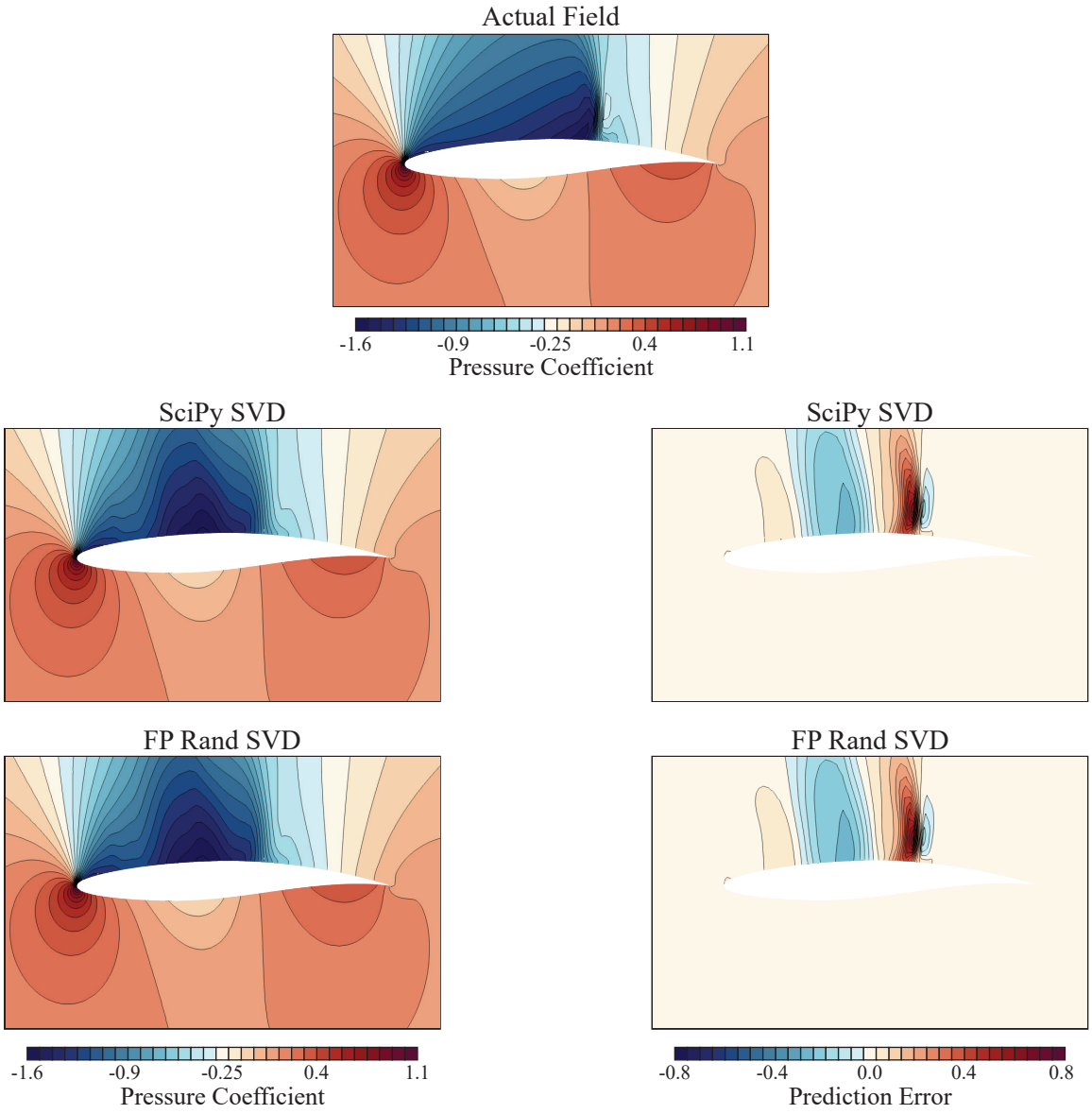


Figure 62: Visual Comparison for Poor Test Point w/ Shock

Figure 60 shows rather surprisingly that the predictive performance of the ROMs using FP-rSVD and SVD is nearly identical. This is perhaps because the approximate random basis is highly accurate as a result of the aggressive values of acceptable error tolerance for the FP-rSVD algorithm. Figure 61 affirms this observation. Even the spread of SMAE for the test points is almost identical for ROMS using both dimension reduction methods. For the sake of completeness, the worst predicted test point (index 13) is visualized in figure 62. It clearly shows that the predictions cannot indeed be told apart visually. In fact, even the error contours are nearly identical. This example highlights the inability of POD and interpolation-based ROMs to capture and predict shocks. The results show compelling evidence that fixed-precision SVD can indeed replace the SVD for large practical problems.

Results for the CRM Wing Problem. Recall that among the problems considered in this research area, the CRM wing problem is the most challenging to tackle because it is 3-dimensional and has complex flow features containing discontinuities. The dimensionality of this flow problem represents scenarios where the whole snapshot matrix struggles to fit in memory on personal computers. In comparison to the airfoil problem, the CRM wing's pressure field exhibits a longer delay in the onset of the flat portion in its singular value decay spectrum. An RIC of 0.9999 results in an actual truncation rank $k = 90$ which is still low-rank in view of the size of the field variable ($\approx 450,000$). But visual inspection of figure 63

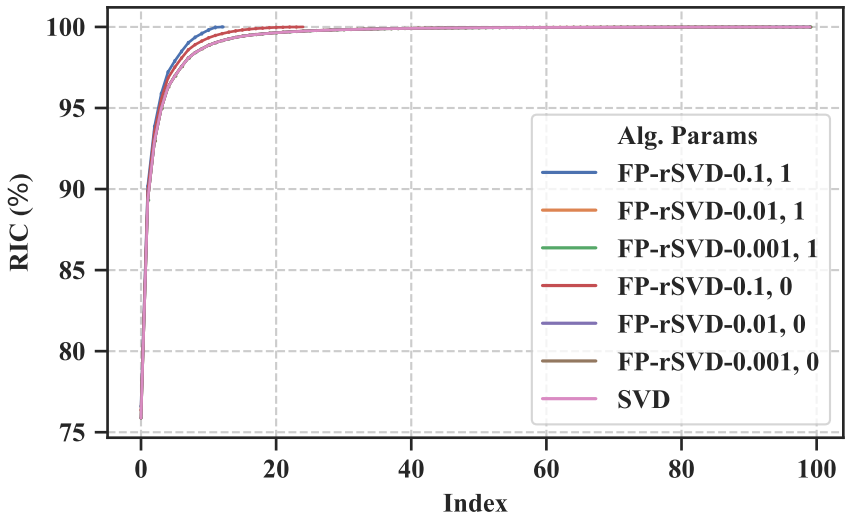
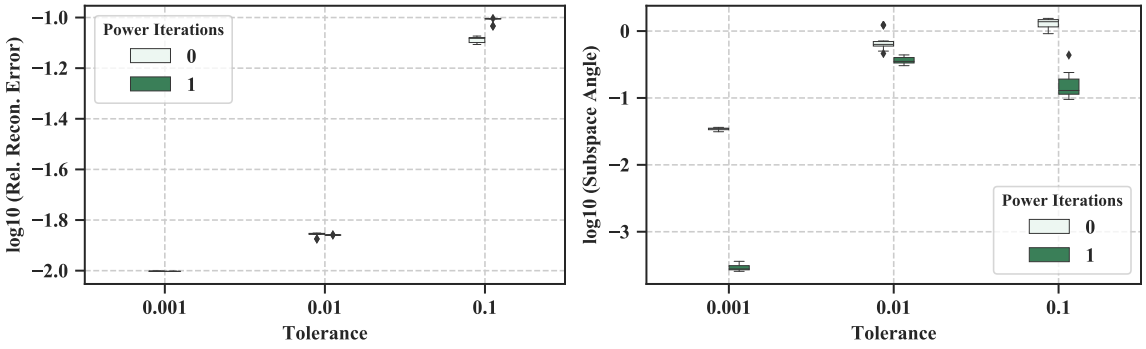


Figure 63: Comparison of RIC for the CRM Wing Problem

shows that the majority of the flow features are captured within indices approximately between 20 and 30. The trailing flat portion responsible for capturing the minor variations in the physical features constitutes a majority of the POD basis vectors. This is an indicator of richness in the physical flow features. As expected therefore, the power iteration parameter (helps with slowly decaying spectrum) has a significant effect on the accuracy of the low-rank approximation for this problem. Figure 63 clearly shows that an $\epsilon = 0.1$ results in a significant inaccuracy in the approximation of the numerical rank and as a result the approximate truncation rank $\tilde{k}$ for an RIC of 0.9999. In fact, even smaller values of ϵ with power iterations resulted in a $\tilde{k}$ much different than k because a majority of the spectrum is characterized by a slow flat decay. Fortunately however, on closer inspection it is clear that beyond an index of approximately 40, the RIC increases marginally. Therefore, as long as the FP-rSVD computes $\tilde{k} \geq 40$ with a good quality basis set, the resulting ROM should exhibit competitive predictive performance.



(a) (b)

Figure 64: Fixed-Precision Randomized SVD: Sensitivity of Projection Error and Quality of Basis for the CRM Wing Problem

Figure 64a clearly shows the effect of the trailing flat singular value spectrum. Notice that in going from $\epsilon = 0.1$ to $\epsilon = 0.01$, there is a large improvement in the reconstruction error. It is in fact nearly identical to ϵ itself. However, when the acceptable error tolerance is decreased to $\epsilon = 0.001$, the reconstruction error cannot be improved beyond 0.01 even with a couple of power iterations. Additionally, the achieved reconstruction error is an order of magnitude larger than ϵ, indicating that the resulting $\tilde{k}$ ignores the last few small singular values. Also notice that the range of acceptable error tolerances for which the

FP-rSVD is tested for the CRM wing problem is an order of magnitude higher than that for the airfoil and canonical problems. As the size of the matrix becomes larger and the information contained in it becomes richer, the FP-rSVD starts demanding a larger number of iterations to converge and therefore quickly starts consuming large amounts of memory to store $\boldsymbol{Q}$ and $\boldsymbol{B}$.

Another strong indicator of how challenging this problem is can be seen in the discrepancy between the random and actual subspaces as measured by the largest subspace angle, as shown in figure 64b. While the power iterations significantly help in achieving small angles, even the smallest ϵ has a largest angle 7 orders of magnitude larger than what was observed in the fixed-rank case (see fig. 46 and fig. 45). However, the discrepancy at high values of ϵ occurs due to the inaccuracies in the small singular values in latter portion of the spectrum. As long as $\tilde{k}$ is comfortably greater than 40 and the trailing insignificant modes contribute to the discrepancy between the actual and approximate subspaces, the effect on the predictive performance of the ROMs should be insignificant.

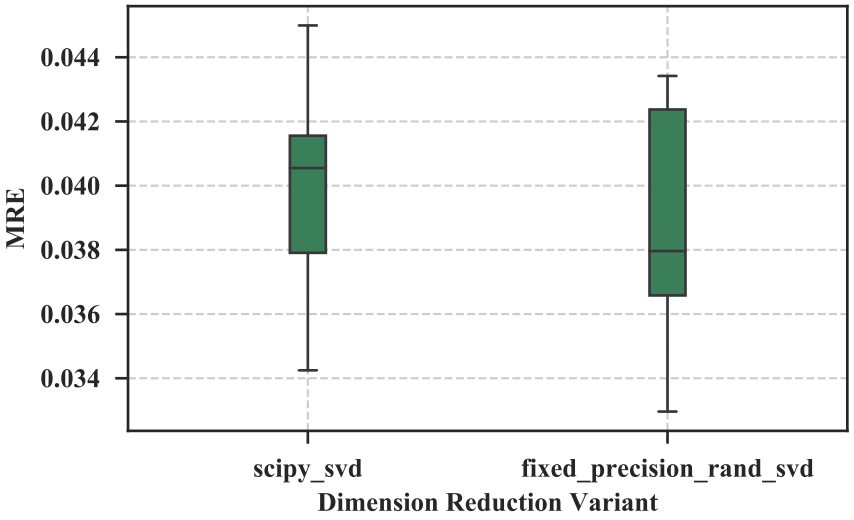


Figure 65: Mean Relative Errors in the Test Set for the CRM Wing Problem

The ROMs using the FP-rSVD and the SVD methods were constructed on 10 randomly generated training sets for the CRM wing problem. The FP-rSVD was run with $\epsilon = 0.01$, 1 power iteration, and block size $b = 10$. This specific value of ϵ consistently resulted in a $\tilde{k}$ of 73, which is different from the actual truncation rank of 90. While power iterations only

marginally improve the largest subspace angle, it was deliberately set to 1 because its value affects the consistency in the computation of $\tilde{k}$. In agreement with the assertion, figure 65 shows that the MRE for ROMs constructed using the FP-rSVD are indeed competitive with the ones that use the deterministic SVD despite the large difference in the values of the approximate truncation rank $\tilde{k}$ and the actual truncation rank k. Although the results for the ROMs using the SVD are marginally more consistent, the difference bears no critical consequence on the magnitude of the MREs. Figure 66 reinforces this observation. The spread of SMAEs for all the points in the test set are similar and the minor differences are

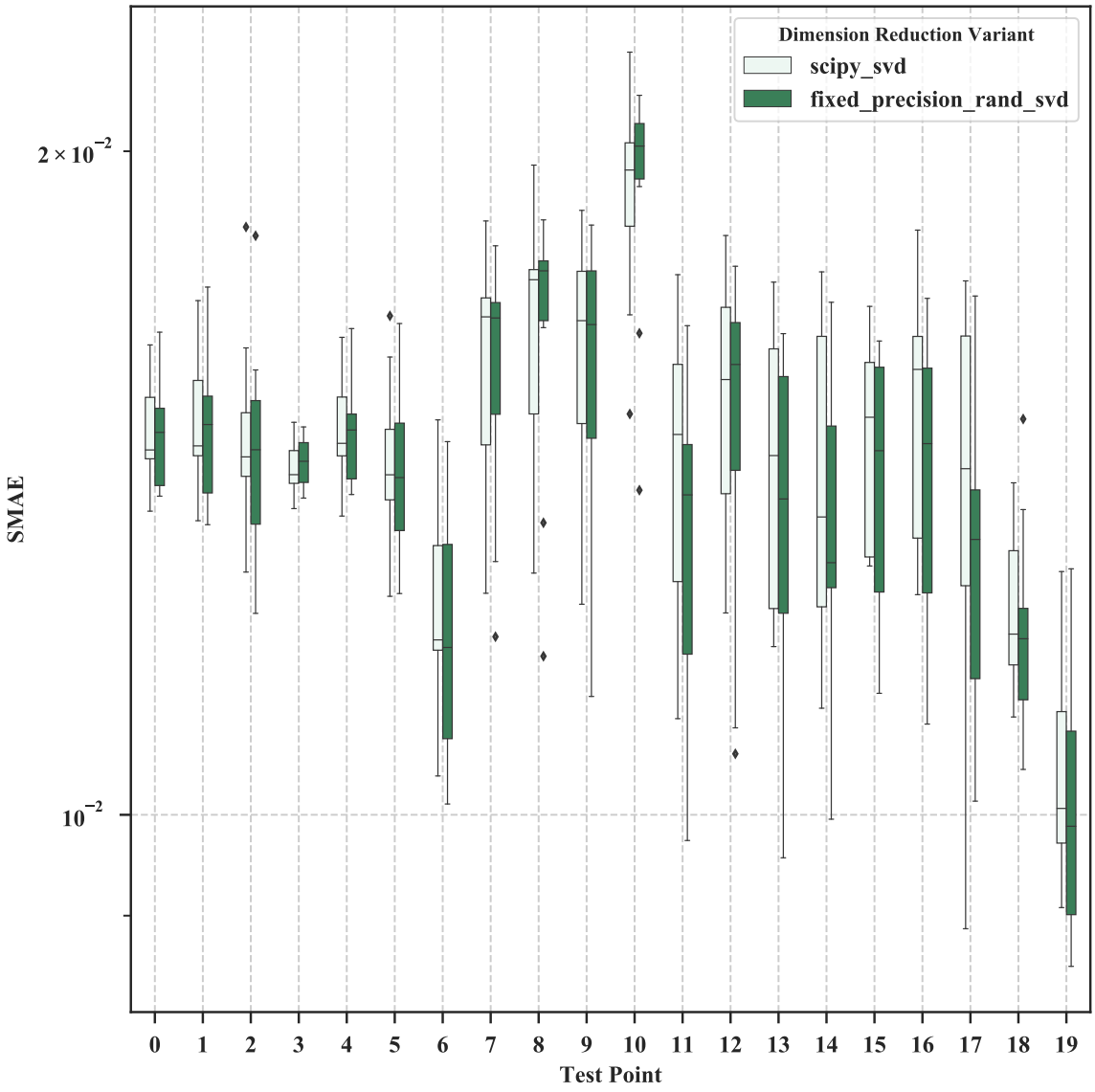


Figure 66: Spatial Mean Absolute Error in the Test Set for the CRM Wing Problem

purely due to the random nature of the FP-rSVD. For the sake of brevity and the fact that all the results are nearly identical, we do not present a visual comparison of the predicted pressure coefficient for this problem.

4.6 Summary and Concluding Remarks

In this chapter, we thoroughly investigated the benefit and effectiveness of randomization as a means to decrease the computational complexity of the dimension reduction step when dealing with large snapshot matrices that may not fit in fast memory. All the experiments and accompanying findings were demonstrated on a canonical problem and two practical flow problems. The study began with the use of fixed-rank algorithms where it was found that given a truncation rank, the rSVD and sketching-based rSVD were both equally effective in computing basis vectors similar to the actual POD basis when run with reasonable values for the algorithm-specific parameters. They did so while significantly reducing the computational complexity of computing the POD. The algorithm-specific parameters provided a means to trade the quality of the POD for gains in computational complexity. Through demonstration on problems with progressively increasing levels of complexity, it was found that large problems with rich physical features generally require relatively conservative values for the algorithm-specific parameters, which limit the potential gains in compute time. Nevertheless, the ROMs using the fixed-rank randomized algorithms were competitive in terms of their predictive accuracy to the benchmark POD and interpolation with the deterministic SVD.

Because specification of the truncation rank a priori is a rather harsh requirement, the second part of this chapter investigated a recently proposed fixed-precision algorithm as a candidate to replace the need to require the truncation rank for computing the decomposition. The FP-rSVD algorithm proved effective in finding the approximate numerical rank given an acceptable error tolerance. We argued that while the error tolerance is analogous to the fixed truncation rank, it is a more intuitive parameter to set. In fact, the results showed that the algorithm was effective in finding the true numerical rank and the actual truncation rank for the canonical problem given a sufficiently small acceptable error tolerance. In

general, it was observed that as the problems became progressively challenging and large, the FP-rSVD required a higher number of iterations to find the numerical rank. Moreover, this approximate numerical rank led to inaccurate values of the approximate truncation rank for a given RIC. Fortunately however, for low-rank problems characterized by a small rapid initial drop in the singular value spectrum followed by a relatively slow flat decay, the FP-rSVD algorithm made errors in the less significant trailing singular values and vectors. Consequently, the resulting ROMs are competitive in their predictive accuracy as long as the singular vectors or POD modes associated with the rapidly decaying singular values are accurately captured. The evidence collected through applications on problems with varying sizes and complexities have empirically shown that techniques from random linear algebra can indeed address and efficiently tackle the challenges posed by large datasets to the *offline* cost, thereby validating our hypothesis which is restated below.

Hypothesis 1: *Techniques from Randomized Linear Algebra (RandNLA) will effectively decrease the computational cost associated with the dimension reduction step while incurring a reasonable penalty in terms of accuracy, thereby enabling efficient construction of parametric ROMs for large systems.*

4.6.1 Steps for Constructing POD and Interpolation-Based ROMs using Randomized POD

The techniques investigated in this research area can be used to replace the POD step with efficient yet accurate randomized algorithms to construct ROMs. Since the focus was on the POD and interpolation-based ROM, we now provide a step-by-step walk through to leverage randomized algorithms depending on the specific application. Let us now look at the details of each step of the methodology. The discussion that follows is summarized in figure 67.

Step 1: Computing the POD Modes via Randomized Methods Following the creation of a design of experiments and evaluation of the full order model, the snapshot matrix $\boldsymbol{W}$ is compressed using either fixed-rank or fixed-precision SVD algorithms to obtain

the POD modes. If the truncation rank is known or the desire is to construct a rank k model, it is recommended to use the fixed-rank algorithm. In the more realistic case that the truncation rank is unknown a priori, it is recommended to use the fixed-precision algorithm to find the approximate truncation rank $\tilde{k}$ given an RIC. Note that in contrast to the deterministic SVD algorithm, an acceptable error tolerance ϵ must also be specified for the fixed-precision algorithm. The specific tolerance that yields a good approximation for the ROM depends on the size of the matrix and its true numerical rank. Because physics-based systems are typically low-rank, even relatively larger error tolerances were found to give ROMs with sufficient predictive accuracy. Performing the SVD using any of the randomized algorithms introduced above will yield

$$\boldsymbol{W} \approx \tilde{\boldsymbol{\Phi}}_k \tilde{\boldsymbol{\Sigma}}_k \tilde{\boldsymbol{V}}_k^T \tag{62}$$

where $\tilde{\boldsymbol{\Phi}}_k \in \mathbb{R}^{m\times k}$, $\tilde{\boldsymbol{\Sigma}}_k \in \mathbb{R}^{k\times k}$, and $\tilde{\boldsymbol{V}}_k \in \mathbb{R}^{n\times k}$ such that $k \ll m$. This POD decomposition approximately minimizes the reconstruction error in the L_2 norm such that

$$\sum_{i=1}^{n} \|\boldsymbol{w}_i - \tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{\Phi}}^T \boldsymbol{w}_i\|_2^2 \approx \sum_{j=k+1}^{\min(m,n)} \tilde{\sigma}_j^2 \tag{63}$$

In the equations above, $\tilde{(\cdot)}$ denotes an approximate quantity as a result of the randomization in the SVD computation process. Success of any dimension reduction-based exercise hinges on one important assumption. It is always assumed that a POD plus interpolation based ROM of high quality can be created for the problems being considered i.e. the problem is low-rank.

Step 2: Expressing Solution in the POD Basis The POD modes thus obtained are assumed to form a physics-based basis set that can be used to approximate any solution as

$$\boldsymbol{w}(\boldsymbol{\mu}) - \bar{\boldsymbol{w}}(\boldsymbol{\mu}) = \sum_{j=1}^{k} a_j(\boldsymbol{\mu})\tilde{\boldsymbol{\phi}}_j \tag{64}$$

$$a_j(\boldsymbol{\mu}) = \langle \boldsymbol{w}(\boldsymbol{\mu}) - \bar{\boldsymbol{w}}(\boldsymbol{\mu}), \tilde{\boldsymbol{\phi}}_j \rangle = \langle \boldsymbol{w}'(\boldsymbol{\mu}), \tilde{\boldsymbol{\phi}}_j \rangle \tag{65}$$

whereby the construction $\bar{\boldsymbol{w}}(\boldsymbol{\mu})$ even allows for the encoding of boundary conditions and other physical constraints [242]. For all the examples in this work, it is taken as the mean

of all the full order solutions in the training data set. Once the POD-basis is computed, projected coordinates in the POD subspace (also called latent space coordinates) are obtained using equation (65) for all the snapshots in $\boldsymbol{W}$ to get $\boldsymbol{W}_r$ (please refer to figure 67).

Step 3: Train Surrogates in the Latent Space via Supervised Learning The last step of the training process involves the creation of surrogate models treating each of the k expansion coefficients as scalar functions of the parameters. Any appropriate supervised learning technique can be employed to create this surrogate model. The work for this research area employs radial basis functions and Gaussian process regression to estimate the expansion coefficient $\boldsymbol{a}(\boldsymbol{\mu}) = \boldsymbol{g}(\boldsymbol{\mu};\ \boldsymbol{W}_r)$ where $\boldsymbol{g} : \mathbb{R}^d \to \mathbb{R}^k$ is the learned map between the parameters and the input space. However, more advanced techniques such as Deep Gaussian Processes (DGP) [207] that can handle non-stationarity [101] and work well in the small data domain can also be employed. Using advanced surrogate models can especially help in capturing highly nonlinear variation of the POD coordinates.

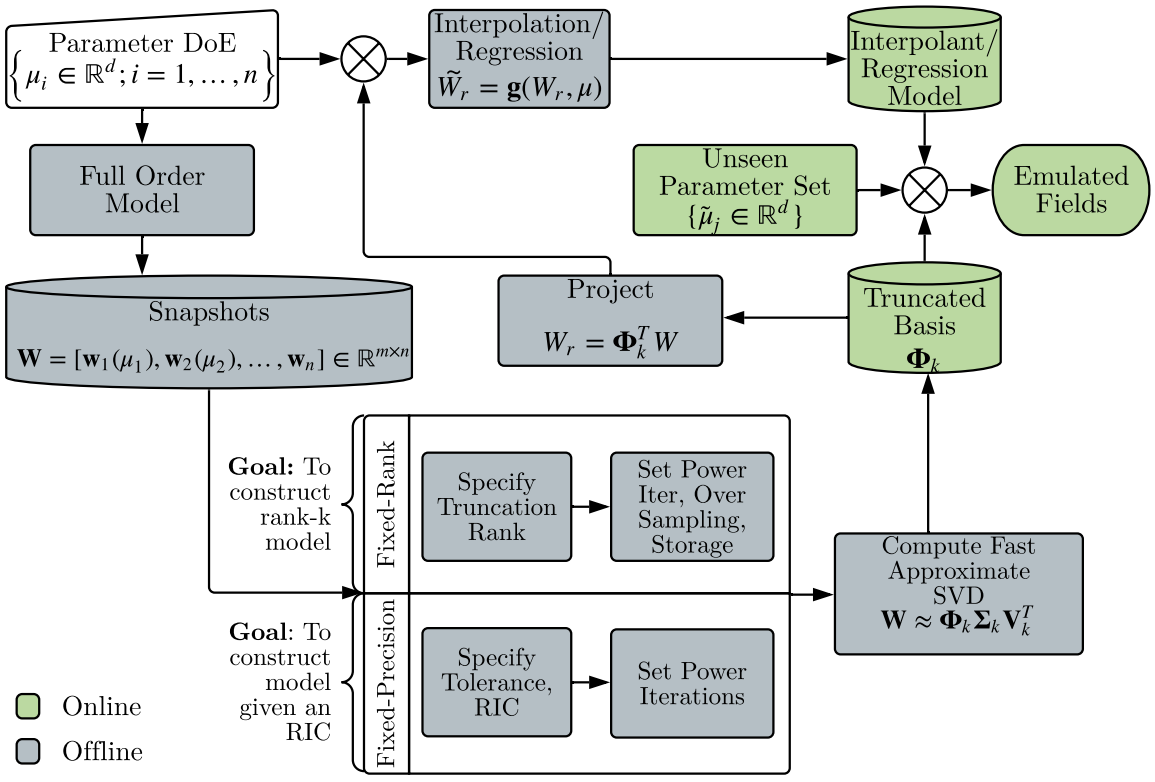


Figure 67: POD and Interpolation using Fixed-Rank and Fixed-Precision Algorithms

Step 4: Emulate Field at Unseen Parameter Point Once the ROM is trained in steps 1-3, the following expression approximates the field at an unseen parameter point $\tilde{\boldsymbol{\mu}}$ in almost real-time.

$$\boldsymbol{w}(\tilde{\boldsymbol{\mu}}) = \bar{\boldsymbol{w}}(\boldsymbol{\mu}) + \tilde{\boldsymbol{\Phi}}\boldsymbol{g}(\tilde{\boldsymbol{\mu}},\ \boldsymbol{W}_r) \tag{66}$$

where $\tilde{\boldsymbol{\Phi}} = [\tilde{\boldsymbol{\phi}}_1, \tilde{\boldsymbol{\phi}}_2, \ldots, \tilde{\boldsymbol{\phi}}_k] \in \mathbb{R}^{m \times k}$ denotes the POD basis.

4.6.2 Comparison of Wall-Time(s)

This final section presents a comparison of the wall-time on varying sizes of the canonical problem. This plot was generated by measuring wall-times for a mix of the algorithm-specific parameters for each problem size over 20 repetitions. The snapshot matrices each had 500 columns and varying row sizes as shown on the x-axis. Despite not focusing on optimized implementations of the various randomized algorithms, figure 68 shows that as expected the rSVD is the most efficient for all problem sizes because of the least number of additional calculations it requires when compared with the other randomized schemes. The sketching-based randomized SVD is slightly more expensive than the FP-rSVD because of

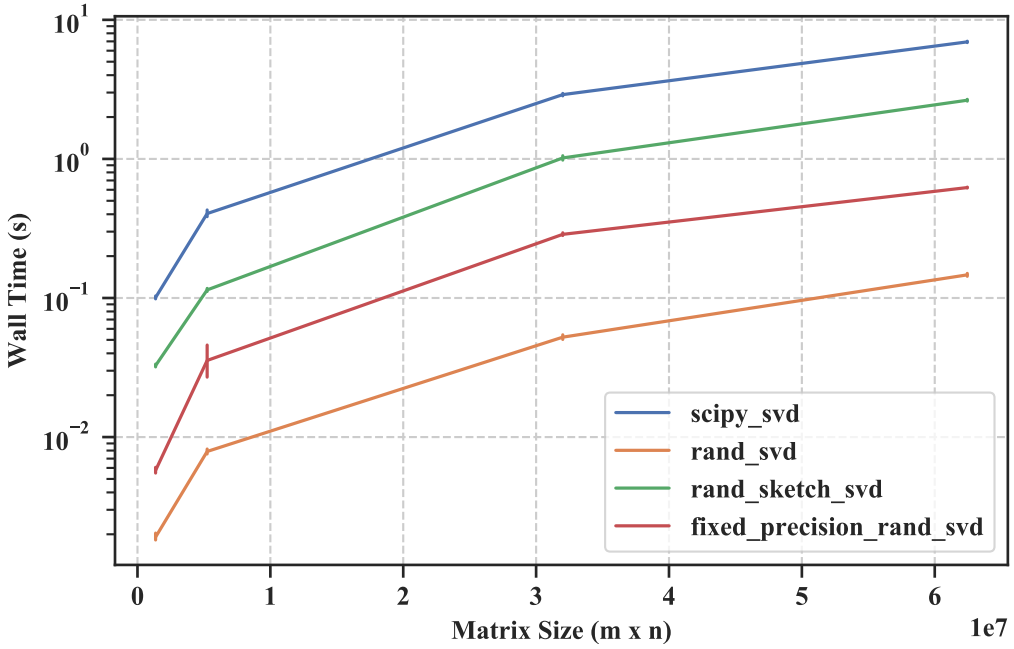


Figure 68: Variation of Wall-Time with Snapshot Matrix Size for the Canonical Problem

the additional overhead introduced by matrix-matrix products to extract the three sketches of the input matrix.

CHAPTER V

POD AND INTERPOLATION VIA GRADIENT-FREE DISCOVERY OF ORTHOGONAL INPUT SUBSPACES USING GAUSSIAN PROCESS REGRESSION IN THE GRASSMANN MANIFOLD

This chapter presents the work done under the second research area presented in chapter 3. Figure 69 shows an overview of the key issue along with an outline of the order of presentation this chapter follows. After providing a thorough literature review of contemporary methods and identifying issues that have received limited attention, we will establish the need and present the development of a novel manifold optimization-based GPR model to tackle large input spaces. Then, we will propose its use to construct POD and interpolation-based ROMs. Finally, the effectiveness of the overall methodology will be demonstrated on applications ranging from a canonical problem with 100 input variables to emulation of the pressure coefficient over a wing with up to 24 input variables.

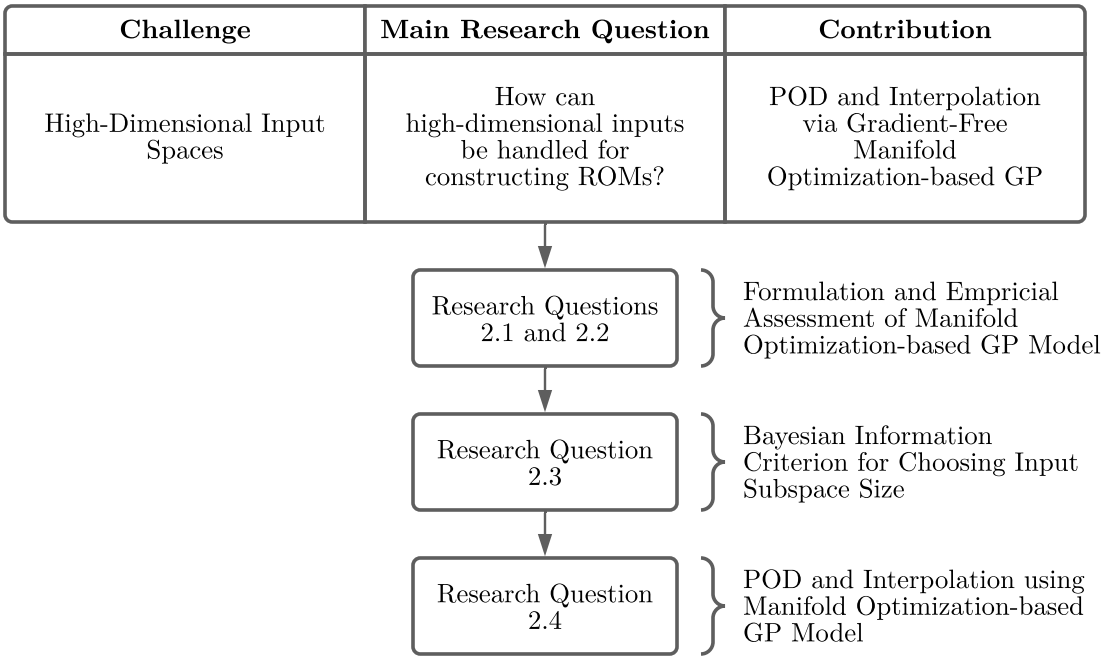


Figure 69: Chapter Outline

In chapter 2, we asserted that within the mapping-based strategies, the projection-based methods are the most promising to construct surrogates for high-dimensional problems. Let us present some arguments to support this assertion. As mentioned before, most of the challenges when tackling a large number of inputs can be directly linked to the *curse of dimensionality* [31,236,255]. Recall that some of the common challenges are: 1) the sample size required for a dense coverage of the input space increases drastically with the number of input parameters, 2) the interpolation for local quantities in higher dimensions can be challenging, and 3) the significance of input parameters on outputs (high-dimensional, as for ROMs) is unknown a priori. Two key illustrations provide concrete substance to these points. First, consider a full-factorial sampling for continuous independent input variables in a d-dimensional unit-hypercube. Generating z^d samples using z discrete levels essentially means dividing the domain into hypercubes with a side length of $1/z$. Because the neighboring cubes share at least one vertex with each other and the volume of the hypercube is $1/z^d$, the distance to any one of these neighbors varies as $\sqrt{d}/z$. As the number of dimensions d increases the volume decreases dramatically whereas the distance between neighbors increases. Therefore, the number of samples to ensure a dense coverage of domains in higher dimensions grows exponentially. Second, [115] discusses an interesting aspect regarding the effect of input space dimensionality in the context of approximating functions using points in local neighborhoods. The statement involves the rate of growth of the volume of the hypercube required to capture a given fraction of all observations. It states that as the number of dimensions increases, the side length of the hypercube quickly increases to an extent such that any pair of points within the so-called neighborhood are too far from each other. Both these examples show that any attempt at devising a dense sampling will most likely be sparse and models that rely on the notion of distance will struggle. Mapping-based techniques are the most suitable for pursuing surrogate modeling in high-dimensions because they directly address the aforementioned issues by creating a map between high- and low-dimensional subspaces. Once, this map is constructed, the surrogate modeling activities are performed in the more manageable low-dimensional subspace.

Remark. The number of repetitions for all the experiments in this chapter was set considering the available computational resources and to reveal any issues regarding bias and/or inconsistency in convergence. Beyond the reported number of repetitions for each experiment, the distribution of the results did not significantly change the observations and conclusions.

5.1 Methods to Construct Surrogates in High-Dimensional Input Spaces

As one may have already observed, the treatment of a large number of inputs using mapping-based methods can be naturally separated into two distinct steps: 1) the computation of a map that projects high-dimensional inputs to a low-dimensional subspace and 2) the training of a surrogate model. To refine our literature search, let us first present some arguments and make choices regarding the kind of surrogate model most suitable for our goals. *Data-fit* surrogate models introduced in chapter 1 can be further classified into parametric and non-parametric models.

The term parametric is loosely used to categorize models that assume a hypothesis space defined using parametrized basis functions. The basis functions when combined linearly result in linear parametric models. To model more complex functions, the basis functions may also be combined in a nonlinear fashion. Artificial Neural Networks (ANNs) are a prime example of such nonlinear and complex models. They are known be exceptionally flexible given an extremely large dataset. The difficulty in selecting the right set of basis functions in a principled manner and the limited predictive power due to a fixed basis function set make this set of methods less appealing for this work. In fact when modeling functions in a high-dimensional setting with limited data when compared with the number of model parameters, this class of models is prone to over-fitting [115]. They are called parametric because the training procedure involves computing the values of a known number of parameters.

On the other hand, non-parametric models make use of the entire dataset to learn the behavior of the underlying function and predict its value given an unseen point. They are typically based on the loose assumption that points that are close in the input space have related function values, i.e. they use the notion of distance between points in the domain

to model the output's variation. For this very reason they are typically ineffective for high-dimensional problems (because of the aforementioned issues related to using distances in high dimensional spaces). K-nearest neighbors and Gaussian Process regression are examples of models that fall under this category. These models do not assume any particular functional form, are less prone to overfitting, and work well with relatively smaller datasets when compared to parametric models.

Parametric models enjoy the advantage of being fast at the cost of making strong assumptions about the data, whereas non-parametric methods pay the price of being computationally intractable with large datasets for enjoying more flexibility. Since non-parametric models offer more flexibility and work well with smaller datasets typically found in engineering design, this dissertation focuses on them. Among the non-parametric supervised learning method types, we will specifically focus on the low-dimensional kind because it is assumed that the high-dimensional inputs will be mapped into a low-dimensional subspace.

Non-parametric low-dimensional models are classified into additive models that assume that the function can be written as a sum of independent additive component functions that each depend only on a combination of a small subset of the original high-dimensional inputs. Note that additive methods operate on the full input space, which can easily increase the number of component functions even for say, a 10-dimensional function if one considers interactions between pairs of inputs. Moreover, whether or not a function is additive is impossible to know a priori.

One of the earliest methods that combined the ideas of mapping and additive models is the famous Projection Pursuit Regression (PPR). The PPR method attempts to find $q \ll d$ directions such that $f(\boldsymbol{\mu}) \approx g_1\left(\boldsymbol{\theta}_1^T\boldsymbol{\mu}\right) + g_2\left(\boldsymbol{\theta}_2^T\boldsymbol{\mu}\right) + \cdots + g_m\left(\boldsymbol{\theta}_m^T\boldsymbol{\mu}\right)$ where $\boldsymbol{\theta}_i \in \mathbb{R}^d$. Although the PPR method enjoys desirable approximation properties, it is costly because it entails training several functions g_i in the original high-dimensional space.

Another previously introduced famous non-parametric surrogate modeling method is the Gaussian Process regression (GPR). GPR models are commonly used for data-fit surrogate modeling with data from computer experiments. They are very flexible interpolators that provide an estimate of the epistemic uncertainty (or predictive uncertainty) in the output.

Although computer experiments are not noisy by definition, we will see how the ability of GPs to model noise is beneficial when used with mapping-based surrogate modeling in high-dimensional spaces.

Now that we have narrowed our choices in terms of the surrogate model itself, let us shift our focus on a state-of-the-art method that computes the dimension reduction map.

5.1.1 Using Active Subspaces for Constructing Surrogate Models

Active subspaces are defined as the leading eigenspaces of the second moment matrix of the function's gradient [277]. They are a generalization of gradient-based GSA in the sense that the eigenspaces need not necessarily be aligned with the canonical coordinates. Since we compare the performance of our proposed model with the ASM, an overview is given below (closely following the textbook by Constantine et. al [72]).

Let $f : \mathbb{R}^d \to \mathbb{R}$ denote a scalar valued function defined over a d-dimensional parameter space (contained in $\mathbb{R}^d$) described by the vector $\boldsymbol{\mu} = [\mu_1, \mu_2, \ldots, \mu_d]^T \in \boldsymbol{M} \subseteq \mathbb{R}^d$. The method further assumes that the derivative of f exists and it is square integrable. Consider the second-moment symmetric positive definite matrix $\boldsymbol{C} \in \mathbb{R}^{d\times d}$ and its eigenvalue decomposition

$$\boldsymbol{C} = \int \nabla f \nabla f^T \rho(\boldsymbol{\mu}) d\boldsymbol{\mu} = \boldsymbol{\Xi}\boldsymbol{\Lambda}\boldsymbol{\Xi}, \ \boldsymbol{\Xi} \in \mathbb{R}^{d\times d}, \text{ and } \boldsymbol{\Lambda} = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_d) \tag{67}$$

where the gradient

$$\nabla f = \left[\frac{\partial f}{\partial \mu_1}, \frac{\partial f}{\partial \mu_2}, \ldots, \frac{\partial f}{\partial \mu_d}\right]^T \in \mathbb{R}^d$$

is a tall vector and ρ denotes a probability measure (may be interpreted as a weighting on input combinations) of choice over the parameter space such that $\rho : \mathbb{R}^m \to \mathbb{R}_+$ and $\int \rho(\boldsymbol{\mu}) d\boldsymbol{\mu} = 1$. In engineering applications, a common choice for ρ is the uniform distribution. Once $\boldsymbol{\mu}$ is viewed as a random variable drawn from ρ, $\boldsymbol{C}$ becomes the expected value of the transformation given by the gradient $\nabla_\mu f : \mathbb{R}^d \to \mathbb{R}^d$, i.e. $\boldsymbol{\mu} \mapsto \nabla f(\boldsymbol{\mu})$. Equation (67) is referred to as the average outer product of the gradient. The i^{th} mean directional derivative of f is given by

$$\lambda_i = \mathbb{E}[(\boldsymbol{\xi}_i^T \nabla f)^2] \tag{68}$$

where $\boldsymbol{\xi}_i \in \mathbb{R}^d$ is the i^{th} column of $\boldsymbol{\Xi}$. A natural ordering of the eigenvalues $\lambda_j \geq \lambda_{j+1} \ \forall \ 1 \leq j \leq d$ permits an intuitive definition and subsequent rank ordering of directions $\boldsymbol{\xi_j}$ such that for larger eigenvalues, the function varies relatively more on average along the corresponding eigenvectors. In other words, directions associated with large eigenvalues carry a higher importance as far as the variation of f is concerned.

Suppose, for the sake of understanding, that $\text{rank}(\boldsymbol{C}) = q < d$ and that the $(d-q)$ trailing eigenvalues are exactly zero. In such cases, f can exactly be represented by a function $g : \mathbb{R}^q \to \mathbb{R}$ because f does not vary along the directions $\boldsymbol{\Xi}_{d-q} = [\boldsymbol{\xi}_{q+1}, \boldsymbol{\xi}_{q+2}, \ldots, \boldsymbol{\xi}_d]$. The resulting function g is called a *ridge function*

$$f(\boldsymbol{\mu}) = g(\boldsymbol{\Xi}_q^T \boldsymbol{\mu}) \tag{69}$$

where $\boldsymbol{\Xi}_q$ is the active subspace consisting of the first q columns of $\boldsymbol{\Xi}$. The matrix $\boldsymbol{\Xi}$ can be partitioned into an active ($[\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_q]$) and an inactive ($[\boldsymbol{\xi}_{q+1}, \boldsymbol{\xi}_{q+2}, \ldots, \boldsymbol{\xi}_d]$) subspace as $\boldsymbol{\Xi} = [\boldsymbol{\Xi}_q \ \ \boldsymbol{\Xi}_{d-q}]$. For the cases in this dissertation and in most simulation models, the trailing eigenvalues are seldom equal to zero but often sum up to a small number. In such situations, the so-called *ridge approximation* is given by

$$f(\boldsymbol{\mu}) \approx g(\boldsymbol{\Xi}_q^T \boldsymbol{\mu}) \tag{70}$$

It can be shown that the approximation error can be bounded by a constant that depends on ρ as

$$||f(\boldsymbol{\mu}) - g(\boldsymbol{\Xi}_q^T \boldsymbol{\mu})||_{L^2_\rho} \leq c_\rho (\lambda_{q+1} + \lambda_{q+2}, \ldots, \lambda_d)^{0.5}$$

where $c_\rho \in \mathbb{R}_+$ is the Poincaré constant. The active and inactive subspaces can be used to define subspaces $\mathcal{X}$ and $\mathcal{Y}$ respectively as

$$\mathcal{X} : \{\boldsymbol{x} \in \mathbb{R}^q \ : \ \boldsymbol{x} = \boldsymbol{\Xi}_q^T \boldsymbol{\mu}\} \ \ \forall \ \ \boldsymbol{\mu} \in \boldsymbol{M} \text{ and } \mathcal{Y} : \{\boldsymbol{y} \in \mathbb{R}^{d-q} \ : \ \boldsymbol{\Xi}_q \boldsymbol{x} + \boldsymbol{\Xi}_{d-q} \boldsymbol{y} \in \boldsymbol{M}\} \ \ \forall \ \ \boldsymbol{x} \in \mathcal{X} \tag{71}$$

The construction of ridge approximations in the active subspace must respect a subtlety, i.e., note that any point $\boldsymbol{x} \in \mathcal{X}$ has at least one corresponding $\boldsymbol{\mu} \in \boldsymbol{M}$, but may have infinitely many $\boldsymbol{\mu}$ because the mapping $\boldsymbol{x} \mapsto \boldsymbol{\mu}$ is ill-posed. If parameter $\boldsymbol{\mu}_i$ is generated by a structured DoE, the projected $\boldsymbol{x}_i$ may not satisfy the same properties. Although the fitted

surrogate assumes random errors in $\boldsymbol{x}_i$, there is indeed structure in the error due to the excluded inactive directions. The general strategy is to build a surrogate by averaging over all $f(\boldsymbol{\mu})$. In practice, for underlying uniform and Gaussian probability distributions, [72,74] give more details on the procedure to construct the surrogate for $g(\boldsymbol{x})$.

Algorithm 6: Basic Procedure to Construct Ridge Approximations for Scalar Valued Functions

Input: n, ρ, f
Output: q, $\boldsymbol{\Xi}_q$, g

1 Draw n i.i.d. samples $\{\boldsymbol{\mu}_i;\ i = 1, 2, \ldots, n\}$ from ρ
2 Evaluate gradient $\nabla f(\boldsymbol{\mu}_i)\ \forall\ i = 1, 2, \ldots, n$
3 Form $\tilde{\boldsymbol{C}} = \frac{1}{n}\sum_{i=1}^{n}(\nabla f(\boldsymbol{\mu}_i))(\nabla f(\boldsymbol{\mu}_i))^T \approx \boldsymbol{C}$ // Approximate second-moment matrix
4 Solve $\tilde{\boldsymbol{C}} = \boldsymbol{\Xi}\boldsymbol{\Lambda}\boldsymbol{\Xi}$ // Diagonalize
/* Inspect gap/jump in $\boldsymbol{\Lambda}$'s spectrum to determine q and choose first-q columns in $\boldsymbol{\Xi}$ (denoted as $\boldsymbol{\Xi}_q$) */
5 Choose $\boldsymbol{\Xi}_q$
6 Compute $\boldsymbol{x}_i = \boldsymbol{\Xi}_q^T\boldsymbol{\mu}_i\ \forall\ i = 1, 2, \ldots, n$ // Collect pairs $(\boldsymbol{x}_i,\ f(\boldsymbol{\mu}_i))$
/* Train Ridge Approximation using collected pairs $(\boldsymbol{x}_i,\ f(\boldsymbol{\mu}_i))\ \forall\ i = 1, 2, \ldots, n$ */
7 Train $\tilde{g}(\boldsymbol{\Xi}_q^T\boldsymbol{\mu}) = f(\boldsymbol{\Xi}_q\boldsymbol{\Xi}_q^T\boldsymbol{\mu}) \approx f(\boldsymbol{\mu})$
8 Return q, $\boldsymbol{\Xi}_q$, $\tilde{g}$

Algorithm 6 shows the stencil to compute the AS and use it to build surrogate models in the projected space. Note that the so-called inactive directions are usually simply discarded. Equation (70) assumes that the trailing eigenvalues are all zero. In reality however, they need not be zero. When the trailing eigenvalues are not zero, the true function can be written as

$$f(\boldsymbol{\mu}) = f(\boldsymbol{\Xi}_q\boldsymbol{x} + \boldsymbol{\Xi}_{d-q}\boldsymbol{y}) \tag{72}$$

Two major points that can have consequences on the surrogate model must be stated here. One, (72) shows that projecting onto the AS and simply ignoring the inactive dimensions has consequences. We will see later that the minor variation in the inactive subspace appears as noise when in the AS. Two, as a result, the actual active subspace dimension q must be chosen. It is typically set by inspecting the spectrum of $\boldsymbol{C}$ and other techniques based on re-sampling have been proposed [72]. However, the choice reduces to how much predictive accuracy one is willing to trade in order to decrease the dimensionality of the input space. Whether or not such a trade is possible is of course an innate property of the function itself.

Before proceeding, let us state the various levels of approximations required when the AS is used for surrogate modeling.

$$\begin{aligned} f(\boldsymbol{\mu}) &= f\left(\boldsymbol{\Xi}_q \boldsymbol{x} + \boldsymbol{\Xi}_{d-q} \boldsymbol{y}\right) \\ &\approx \int f\left(\boldsymbol{\Xi}_q \boldsymbol{x} + \boldsymbol{\Xi}_{d-q} \boldsymbol{y}\right) \pi(\boldsymbol{y} \mid \boldsymbol{x}) d\boldsymbol{y} \\ &= g\left(\boldsymbol{\Xi}_q^T \boldsymbol{\mu}\right) \\ &\approx \hat{g}\left(\boldsymbol{\Xi}_q^T \boldsymbol{\mu}\right) \\ &\approx \tilde{g}\left(\boldsymbol{\Xi}_q^T \boldsymbol{\mu}\right) \\ &\approx \tilde{g}\left(\boldsymbol{\Xi}_q^T \boldsymbol{\mu}\right) \end{aligned} \tag{73}$$

where $\hat{g}$ is the Monte-Carlo approximation of the function g which is obtained by integrating out the inactive subspace and $\tilde{g}$ is the surrogate model. Observe that the ASM relies on the availability of gradients from the simulation model. Most modern computer programs avail access to gradients through automatic differentiation or adjoint formulations. However, this requirement hinders the use of ASM with simulation models (such as legacy codes) that do not provide access to gradients. More importantly, in the context of ROMs, the gradients of either the coordinates in the projected subspace or the entries of the reduced projected operators are required. Their computation entails intrusive modification of the source code. While finite-differences may be employed to approximate gradients, the number of model evaluations scales poorly with a large number of input parameters. One of the desirable features of the method is its ability to find the AS and train a surrogate model using a number of training samples proportional to the logarithm of the number of input dimensions. Therefore, there is a need for a gradient-free method to discover a low-dimensional input subspace that uses a competitive number of training samples to construct a good surrogate in it.

5.1.2 Gradient-Free Alternatives

Learning ridge functions (70) exclusively from point data has a rich body of literature based on approximation theory [69, 196]. Authors in [94] learn the column space of $\boldsymbol{\Xi}$ assuming its basis is sparse. Yet another lens of viewing the subspace estimation problem is through

the theory of matrix completion and recovery [87]. Topics related to Sufficient Dimension Reduction [271] also directly estimate the range of $\boldsymbol{\Xi}$. Recent developments [70, 144] have successfully attempted the construction of the active subspace through Morris' screening procedure. They are generally more effective in recovery when the parameter space is sampled adaptively as the subspace is constructed. In [86], the authors propose a new randomized sampling-based strategy to approximate the second-moment matrix. Gradient-sketching [75] as an approximation to the gradient has also shown success in recovering the active subspace. Note that the methods mentioned above successfully recover the active subspace by approximating the gradient using finite-difference type schemes. Let us present a short summary of two recent strategies that rely on approximating the AS using finite-difference type gradient-sketching to highlight why they are inapplicable for our use case. The first scheme relies on gradient-sketching [75] whereas the second relies on newly published work [86] that provides sound theoretical guarantees with relaxed assumptions on the underlying probability measure ρ. It is worth pointing out that the authors in both [75, 86] show results for scalar-valued outputs.

The first approach attempts to approximate the active subspace from linear measurements of the matrix of gradients. Suppose for the scalar function $h_i(\boldsymbol{\mu})$, the i^{th} component in the latent space defined by the POD basis, the linear measurement $\mathcal{M}(\cdot)$ of n samples of its gradients is given by

$$\mathcal{M}(\boldsymbol{G}) = \begin{bmatrix} \boldsymbol{S}_1^T \nabla h_i(\boldsymbol{\mu}_1) & \boldsymbol{S}_2^T \nabla h_i(\boldsymbol{\mu}_2) & \dots & \boldsymbol{S}_n^T \nabla h_i(\boldsymbol{\mu}_n) \end{bmatrix}, \quad \boldsymbol{S}_j \in \mathbb{R}^{d \times l} \; \forall \; j = 1, 2, \dots, n \tag{74}$$

where $\boldsymbol{S}_j$ is called a sketching matrix. Note that the entries of the measurement matrix may be viewed as an inexpensive approximation of the directional derivative of g or a linear measurement of the gradient. For a small enough $\epsilon > 0$ and direction $\boldsymbol{a}$

$$\boldsymbol{a}^T \nabla h_i(\boldsymbol{\mu}) \approx \frac{h_i(\boldsymbol{\mu} + \epsilon \boldsymbol{a}) - h_i(\boldsymbol{\mu})}{\epsilon} \tag{75}$$

Instead of taking $d+1$ function evaluations, the sketch or linear measurement requires only two function evaluations, i.e., it is independent of the number of input parameters d. Now, consider an analogous measurement matrix for the gradient of a vector-valued $h(\boldsymbol{\mu})$. Given

$h : \mathbb{R}^d \to \mathbb{R}^k$, its Jacobian is defined as

$$\boldsymbol{J}(\boldsymbol{\mu}) = \begin{bmatrix} \vdots & & \vdots \\ \nabla h_1(\boldsymbol{\mu}) & \dots & \nabla h_k(\boldsymbol{\mu}) \\ \vdots & & \vdots \end{bmatrix} \in \mathbb{R}^{d \times k}$$

The measurement operator (as in eq. (74) for scalars) for $\boldsymbol{J}$ can be defined as

$$\mathcal{M}(\boldsymbol{G_J}) = \begin{bmatrix} \boldsymbol{S}_1^T \boldsymbol{J}(\boldsymbol{\mu}_1) & \boldsymbol{S}_2^T \boldsymbol{J}(\boldsymbol{\mu}_2) & \dots & \boldsymbol{S}_n^T \boldsymbol{J}(\boldsymbol{\mu}_n) \end{bmatrix}, \quad \boldsymbol{S}_j \in \mathbb{R}^{d \times l} \ \forall \ j = 1, 2, \dots, n \tag{76}$$

where the entries of $\boldsymbol{S}_i$ are drawn from independent Gaussians. Following [75], an estimate of the active subspace may be obtained by solving a least-squares problem posed as

$$\min_{\boldsymbol{A},\ \boldsymbol{B}} \ ||\mathcal{M}(\cdot) - \mathcal{M}(\boldsymbol{A}\boldsymbol{B}^T)||_F \tag{77}$$

where the argument to $\mathcal{M}(\cdot)$ is either $\boldsymbol{G}$ or $\boldsymbol{G_J}$ depending on whether the problem is scalar- or vector-valued. The alternating least-squares algorithm may be employed to solve the problem in equation (77). Note that for a rank-q approximation of the active subspace, $l > q$, i.e. the number of linear measurements l must be greater than q. Once equation (77) is solved, the active subspace is given by the left singular vectors of $\boldsymbol{A}\boldsymbol{B}^T$.

The second procedure for estimating the active subspace follows the work in [86]. A brief outline of the procedure for scalars is outlined below.

1. Start with N points drawn independently from the underlying probability distribution ρ as $X : \{\boldsymbol{\mu}_j, h_i(\boldsymbol{\mu}_j)\}_{j=1}^N$.

2. Draw $N_{X,\epsilon}$ random samples from the conditional probability distribution $\rho_{X,\epsilon}$ on the union of Euclidean balls $\mathbb{B}_{\boldsymbol{\mu},\epsilon}$ of radius $\epsilon > 0$.

$$\mathbb{B}_{X,\epsilon} = \bigcup_{\boldsymbol{\mu} \in X} \mathbb{B}_{\boldsymbol{\mu},\epsilon} \tag{78}$$

3. Partition the set $Y_{X,\epsilon}$ of additionally sampled points $N_{X,\epsilon}$ according to $Y_{\boldsymbol{\mu},\epsilon} = Y_{X,\epsilon} \cap \mathbb{B}_{\boldsymbol{\mu},\epsilon} \ \ \forall \ \ \boldsymbol{\mu} \in X$.

4. For each $\boldsymbol{\mu} \in X$, compute $\nabla^{Y_{X,\epsilon}} h_i(\boldsymbol{\mu})$ as an estimate of the true gradient using,

$$\nabla^{Y_{X,\epsilon}} h_i(\boldsymbol{\mu}) = \frac{d}{N_{\boldsymbol{\mu},\epsilon}} \sum_{\boldsymbol{y} \in Y_{\boldsymbol{\mu},\epsilon}} \frac{h_i(\boldsymbol{y}) - h_i(\boldsymbol{\mu})}{||\boldsymbol{y} - \boldsymbol{x}||_2} \cdot \frac{\boldsymbol{y} - \boldsymbol{\mu}}{||\boldsymbol{y} - \boldsymbol{x}||_2} \tag{79}$$

5. Finally, find an estimate of the second-moment matrix $\tilde{\boldsymbol{C}}$ as,

$$\tilde{\boldsymbol{C}} = \frac{1}{N} \sum_{\boldsymbol{\mu} \in X} \nabla^{Y_{X,\epsilon}} h_i(\boldsymbol{\mu}) \nabla^{Y_{X,\epsilon}} h_i(\boldsymbol{\mu})^T \tag{80}$$

Note that both the methods presented above rely on a prescribed sampling plan, i.e. their effectiveness relies on a structured stencil to query points in the input domain. On a closer examination, these methods require a dense sampling of the input space because they fundamentally try to approximate gradients in a high-dimensional space using finite-differences. As such, these strategies are ineffective or inapplicable in cases where either the underlying function is computationally expensive to evaluate or the input-output pairs are prescribed a priori. In what follows, an improved method based on optimization of manifolds is proposed. Given input-output pairs, the method trains a Gaussian process with a modified dimension reduction kernel that essentially finds the low-dimensional subspace in conjunction with the hyperparameters to minimize the reconstruction error.

5.1.3 Manifold Optimization-Based Gaussian Process Regression with an Input Space Dimension Reduction Kernel

This section begins with a complete description of a conventional Gaussian process (GP) regression followed by the modification of the kernel to accommodate input space dimension reduction. Then, the details of a novel strategy to optimize for the subspace and the hyperparameters of the kernel are presented. The section concludes with the details of a new gradient-free method to leverage the GPs for construction of POD and interpolation-based ROMs.

5.1.3.1 Conventional Gaussian Process Regression

Let $\boldsymbol{M} : \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_n \in \mathbb{R}^d\}$ be inputs that generate outputs $\boldsymbol{Y} : \{y_1, y_2, \ldots, y_n\}$ under an arbitrary function $h : \mathbb{R}^d \to \mathbb{R}$. In the case of regression, given a set of input-output pairs, a GP updates a prior distribution over functions to a posterior distribution enforcing the assumption that $p(y_1 = h(\boldsymbol{\mu}_1), y_2 = h(\boldsymbol{\mu}_2), \ldots, y_n = h(\boldsymbol{\mu}_n))$ is jointly Gaussian. The joint Gaussian is fully specified using a mean $\boldsymbol{m}(\boldsymbol{\mu})$ and a covariance $\boldsymbol{\Sigma}(\boldsymbol{\mu})$. The kernel $\kappa(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j; \boldsymbol{\theta}) = \Sigma_{ij}$ that makes up the covariance captures the similarity in the function

values for a pair of points using their closeness. Here $\boldsymbol{\theta}$ denotes the parameters of the kernel; called the hyperparameters of the GP.

If the samples of the function are noisy, i.e., given $y = h(\boldsymbol{\mu}) + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma_y^2)$, the covariance becomes $\boldsymbol{\Sigma}_y = \boldsymbol{\Sigma} + \sigma_y^2 \boldsymbol{I}_n$. The Bayes update rule can be used to write the posterior GP as

$$h(\cdot)|\boldsymbol{M}, \boldsymbol{Y}, \boldsymbol{\theta}, \sigma_y \sim GP(h(\cdot)|\tilde{m}(\cdot), \tilde{\kappa}(\cdot, \cdot)) \tag{81}$$

$$\tilde{m}(\boldsymbol{\mu}) = m(\boldsymbol{\mu}) + \boldsymbol{\Sigma}_y(\boldsymbol{\Sigma}_y + \sigma_y^2 \boldsymbol{I}_n)^{-1}(\boldsymbol{y} - \boldsymbol{m}) \tag{82}$$

$$\tilde{\kappa}(\cdot, \prime) = \kappa(\cdot, \prime; \boldsymbol{\theta}) - \boldsymbol{\Sigma}_y(\boldsymbol{\Sigma}_y + \sigma_y^2 \boldsymbol{I}_n)^{-1}\boldsymbol{\Sigma}_y \tag{83}$$

Under the Bayesian formalism, beliefs about the underlying function are imposed through the choice of the mean, the kernel function and a prior distribution for the kernel hyperparameters written as

$$\boldsymbol{\theta} \sim p(\boldsymbol{\theta}) \tag{84}$$

The posterior over the hyperparameters given data is then given by

$$p(\boldsymbol{\theta}, \sigma_y|\boldsymbol{M}, \boldsymbol{Y}) \propto p(\boldsymbol{y}|\boldsymbol{M}, \boldsymbol{\theta}, \sigma_y)p(\boldsymbol{\theta})p(\sigma_y) \tag{85}$$

Equations (81) and (85) together give a complete description of the information about the underlying function under the observed data. It is more intuitive and convenient to deal with the so-called predictive probability density conditioned on the hyperparameters

$$h(\boldsymbol{\mu})|\boldsymbol{M}, \boldsymbol{Y}, \boldsymbol{\theta}, \sigma_y \sim \mathcal{N}(h(\boldsymbol{\mu})|\tilde{m}(\boldsymbol{\mu}), \tilde{\kappa}(\boldsymbol{\mu}, \boldsymbol{\mu})) \tag{86}$$

where $\tilde{m}(\boldsymbol{\mu})$ (eq. (81)) and $\tilde{\kappa}(\boldsymbol{\mu}, \boldsymbol{\mu})$ (eq. (83)) are the predictive mean and variance at $\boldsymbol{\mu}$, respectively.

The training procedure ideally characterizes the posterior distribution of the hyperparameters. In this work, a much simpler approach to solve for a point estimate of the hyperparameters by maximizing the likelihood of the observed data is adopted. For issues concerning numerical stability, it is commonplace to maximize the log of the marginal likelihood to train the GPR model.

$$\mathcal{L}(\boldsymbol{\theta}, \sigma_y; \boldsymbol{\mu}, \boldsymbol{y}) = \log p(\boldsymbol{y}|\boldsymbol{\mu}) = \log \mathcal{N}(\boldsymbol{y}|\boldsymbol{0}, \boldsymbol{\Sigma}_{\boldsymbol{y}}) = -\frac{1}{2}\boldsymbol{y}\boldsymbol{\Sigma}_{\boldsymbol{y}}^{-1}\boldsymbol{y} - \frac{1}{2}\log|\boldsymbol{\Sigma}_{\boldsymbol{y}}| - \frac{n}{2}\log 2\pi \tag{87}$$

The hyperparameters that maximize the log marginal likelihood are computed by solving the following optimization problem:

$$\underset{\boldsymbol{\theta},\sigma_y}{\text{minimize}} \quad -\mathcal{L}(\boldsymbol{\theta},\sigma_y;\boldsymbol{M},\boldsymbol{Y})$$

The value of hyperparameters thus obtained is referred to as the maximum likelihood estimate (MLE). It must be noted that the log-marginal likelihood is known to be non-convex. When used with gradient-based optimizers, a common practice is to repeat the optimization multiple times from random initializations of hyperparameter guesses.

5.1.3.2 Gaussian Process with Input Space Dimension Reduction Kernel

As stated previously, an objective of the proposed approach is to uncover a low-dimensional subspace of the original input space in a supervised fashion without resorting to gradient evaluations. Among low-dimensional subspaces of the input space, the active subspace as defined previously exhibits desirable features and therefore appears to be a good candidate. The active subspace is defined as the range of the projection matrix $\boldsymbol{\Xi}_q$. Therefore, knowledge of $\boldsymbol{\Xi}_q$ implies knowledge of the active subspace. As a consequence, previous work on the recovery of the active subspace within a supervised learning training process has focused on approximating the matrix $\boldsymbol{\Xi}_q$ by relying on an approximation model of the form $f(\boldsymbol{\mu}) \approx g(\boldsymbol{\Xi}_q^T\boldsymbol{\mu})$ where g is modeled using a GP and $\tilde{\boldsymbol{\Xi}}$ is meant to approximate $\boldsymbol{\Xi}_q$. Note that $\tilde{\boldsymbol{\Xi}}$ need not necessarily be related to $\boldsymbol{\Xi}_q$. Because the AS exhibits desirable features, it is hoped that $\tilde{\boldsymbol{\Xi}}$ resembles $\boldsymbol{\Xi}_q$. Those approaches are succinctly reviewed and discussed in the following paragraphs, and potential improvement areas are identified, that motivate the new formulation proposed in this dissertation.

In contrast to traditional GPs, when the model is augmented with the projection matrix $\tilde{\boldsymbol{\Xi}}$, the marginal likelihood becomes a function of both the GP hyperparameters and the projection matrix $\tilde{\boldsymbol{\Xi}}$. Consequently, the log marginal likelihood must not only be optimized with respect to the hyperparameters, but also with respect to the projection matrix $\tilde{\boldsymbol{\Xi}}$. The optimization with constraints on the orthonormality of matrices introduces additional challenges.

By construction, $\boldsymbol{\Xi}_q$ is orthonormal and therefore, it is natural to constrain its approximation $\tilde{\boldsymbol{\Xi}}$ to be orthonormal during optimization. In [250], an unconstrained $d \times q$ matrix is transformed into an orthonormal matrix by a computing QR factorization. On the other hand, [249] exploits the concept of the Stiefel manifold which is the set of orthogonal matrices defined as

$$St(q, \mathbb{R}^d) = \{\boldsymbol{X} \in \mathbb{R}^{d \times q} : \boldsymbol{X}^T \boldsymbol{X} = \boldsymbol{I}_p\} \tag{88}$$

During the optimization process, [249] used a Crank-Nicholson type update to ensure $\tilde{\boldsymbol{\Xi}}$ remained in the Stiefel manifold, i.e. the optimization steps were constructed such that every iterate belonged to the Stiefel manifold. This resulted in a monolithic numerical optimization routine for $\tilde{\boldsymbol{\Xi}}$, that was carried out separately from the optimization of the GP hyperparameters. In a two-step approach, the GP hyperparameters and $\tilde{\boldsymbol{\Xi}}$ had to be respectively held constant while the other was being optimized. While both approaches showed promise, they focused on solving for an orthonormal basis $\boldsymbol{\Xi}$ of the active subspace. There are infinitely many such bases. If the end goal is to identify the subspace itself, the choice of a specific basis is inconsequential, and the search can be simplified by exclusively directing focus to linear subspaces instead of orthonormal bases.

Similar to the space of $d \times q$ orthonormal matrices that admits a manifold structure (the Stiefel manifold, eq. (88)), the set of q-dimensional subspaces of $\mathbb{R}^d$ can be endowed with the structure of a smooth compact manifold called the Grassmann manifold denoted by $Gr(q, \mathbb{R}^d)$. As pointed out in [3], the Grassmann manifold is not directly defined in terms of matrices and is therefore harder to visualize. A point in the Grassmann manifold can be conceived as an equivalence class of $d \times q$ matrices; all $d \times q$ matrices with identical column span are represented by a single point in the Grassmann manifold. Parametrizing $\tilde{\boldsymbol{\Xi}}$ using a Grassmann manifold refines the search by focusing exclusively on the subspace rather than the choice of the basis set used to represent the subspace. Although manifolds are abstract concepts, working with them brings practical advantages when it comes to matrix and subspace optimization.

Firstly, the actual dimension of the search space is known, avoiding redundant search

dimensions compared to more naïve approaches. While $d \times q$ projection matrices are represented by $d \times q$ unknowns, the actual dimensions of the Stiefel manifold is in fact lower than $d \times q$:

$$\dim(St(q, \mathbb{R}^d)) = q \times d - \frac{q(q+1)}{2} \tag{89}$$

Since all orthonormal matrices spanning the same subspace are only represented by a single point in the Grassmann manifold, its dimension is even lower than the corresponding Stiefel manifold:

$$\dim(Gr(q, \mathbb{R}^d)) = q \times (d - q) \tag{90}$$

Reducing the number of dimensions of the search space effectively reduces its size and speeds up the optimization.

Secondly, working in differentiable manifolds offers a structured mathematical construct. Rules of calculus analogous to those found in traditional Euclidean spaces can be derived for manifolds. As a result, numerical optimization schemes commonly used in engineering have been developed for manifolds [3]. Practical implementations such as `pymanopt` [247] are made openly available to practitioners. Working in such a framework affords the ability to focus on the formulation of the manifold optimization problem at hand instead of the technical aspects of the numerical solver. In fact, because the formulation of the manifold optimization problem and the solution methods are completely independent, solvers and line search algorithms can conveniently be swapped. User-defined algorithms can also be implemented and used in lieu of those packaged with the library.

Thirdly, optimization on elements of seemingly different mathematical natures, such as GP hyperparameters taking values in $\mathbb{R}$ and linear low-dimensional subspaces belonging to $Gr(q, \mathbb{R}^d)$ can all be treated under the same umbrella using the concept of product manifolds. Instead of performing separate searches in $\mathbb{R}^l$ (where l is the number of GP hyperparameters) and $Gr(q, \mathbb{R}^d)$, a single unified search can then be performed in the product manifold $\mathbb{R}^l \times Gr(q, \mathbb{R}^d)$.

The discussion above encourages and justifies the following proposed model form for the

GP

$$h(\boldsymbol{\mu})|\boldsymbol{\theta}' \sim GP(h(\boldsymbol{\mu})|\kappa_{DR}(\boldsymbol{\mu}, \boldsymbol{\mu}^*; \boldsymbol{\Xi}, \boldsymbol{\theta}')) \tag{91}$$

where

$$\kappa_{DR}(\boldsymbol{\mu}, \boldsymbol{\mu}^*; \boldsymbol{\Xi}, \boldsymbol{\theta}') = \kappa(\boldsymbol{\Xi}^T\boldsymbol{\mu}, \boldsymbol{\Xi}^T\boldsymbol{\mu}^*; \boldsymbol{\theta}') \tag{92}$$

$$\kappa_{DR} : \mathbb{R}^d \times \mathbb{R}^d \times Gr(q, \mathbb{R}^d) \times \mathbb{R}^l \to \mathbb{R} \tag{93}$$

$$\boldsymbol{\mu}, \boldsymbol{\mu}', \boldsymbol{\Xi}, \boldsymbol{\theta}' \to \kappa_{DR}(\boldsymbol{\mu}, \boldsymbol{\mu}', \boldsymbol{\Xi}) \tag{94}$$

and

$$\kappa : \mathbb{R}^q \times \mathbb{R}^q \times \mathbb{R}^l \to \mathbb{R} \tag{95}$$

$$\boldsymbol{\Xi}^T\boldsymbol{\mu}, \boldsymbol{\Xi}^T\boldsymbol{\mu}', \boldsymbol{\theta}' \to \kappa(\boldsymbol{\Xi}^T\boldsymbol{\mu}, \boldsymbol{\Xi}^T\boldsymbol{\mu}', \boldsymbol{\theta}') \tag{96}$$

Here, κ_{DR} is the novel input space dimension reduction kernel and κ is the usual kernel function in the projected input space. The log marginal likelihood for the newly defined kernel can be written as

$$\mathcal{L}(\boldsymbol{\theta}', \sigma_y; \boldsymbol{\mu}, \boldsymbol{y}) = \log p(\boldsymbol{y}|\boldsymbol{\mu}) = \log \mathcal{N}(\boldsymbol{y}|\boldsymbol{0}, \boldsymbol{\Sigma}'_{\boldsymbol{y}}) = -\frac{1}{2}\boldsymbol{y}{\boldsymbol{\Sigma}'_{\boldsymbol{y}}}^{-1}\boldsymbol{y} - \frac{1}{2}\log|\boldsymbol{\Sigma}'_{\boldsymbol{y}}| - \frac{n}{2}\log 2\pi \tag{97}$$

$$\boldsymbol{\Sigma}'_y = \boldsymbol{\Sigma}' + \sigma_y^2\boldsymbol{I}_n,\ \kappa_{DR}(\boldsymbol{\mu}_i, \boldsymbol{\mu}_j; \boldsymbol{\Xi}, \boldsymbol{\theta}') = \Sigma'_{ij} \tag{98}$$

As mentioned above, both the GP hyperparameters $\boldsymbol{\theta}'$ and the projection subspace $\tilde{\boldsymbol{\Xi}}$ are optimized together by posing the optimization problem on the product manifold $\mathbb{R}^{l+1} \times Gr(q, \mathbb{R}^d)$ as follows

$$\underset{(\boldsymbol{\Xi}, \boldsymbol{\theta}', \sigma_y) \in Gr(q, R^d) \times R^{l+1}}{\text{minimize}} \quad -\mathcal{L}(\boldsymbol{\Xi}, \boldsymbol{\theta}', \sigma_y; \boldsymbol{M}, \boldsymbol{Y}) \tag{99}$$

While the method is compatible with any kernel, all the experiments performed in this work use the Automatic Relevance Determination (ARD) squared exponential kernel defined as

$$\kappa_{\text{SE}}(\boldsymbol{\mu}, \boldsymbol{\mu}') = \sigma_s^2 \exp\left(-r^2\right) \tag{100}$$

where r is the distance function given by

$$r = \sqrt{\sum_{i=1}^{d} \frac{(\mu_i - \mu'_i)^2}{2l_k^2}}$$

where $l_i,\ i = 1, \ldots, d$ are the length scales associated with each dimension in $\boldsymbol{\mu}$ and σ_s is the signal variance. Some other popular choices for kernels are the Matérn class of kernels. Popularly found in engineering applications are the 3/2 and the 5/2 kernels for which the expressions are given as

$$\kappa_{3/2}\left(\boldsymbol{\mu}, \boldsymbol{\mu}'\right) = (1 + \sqrt{3}r)\exp(-\sqrt{3}r)$$
$$\kappa_{5/2}\left(\boldsymbol{\mu}, \boldsymbol{\mu}'\right) = \left(1 + \sqrt{5}r + \frac{5}{3}r^2\right)\exp(-\sqrt{5}r)$$

The noise variance σ_y, signal variance σ_s, and length scales $l_i,\ i = 1, \ldots, d$ together constitute the set of hyperparameters of the kernel that live in Euclidean space. For all the experiments, the initial values of both the logarithm of the squared length scales and the logarithm of signal variance are drawn from the uniform distribution between -1 and 10. The logarithm of the noise variance is drawn from a uniform distribution between -10 and 1. Note that posing the problem using the logarithm makes the optimization problem unconstrained by construction. In addition to the hyperparameters, $\boldsymbol{\Xi}$ is the reduced input subspace which belongs to the Grassmann manifold and its initial value is a random sample from the Grassmann manifold.

In the discussion thus far, it has been implicitly assumed that the reduced input space dimension q is known. In reality though, it is unknown since the subspace is estimated as part of the GP training procedure. In this work, the manifold optimization approach is first explored for problems where the actual dimension of the input subspace is known. Later, we will propose the use of the Bayesian Information Criterion (BIC) to determine the dimensionality of the input subspace [249]. From now on, this method will be referred to as the Manifold Optimization-based Gaussian Process (MO-GP) in this dissertation.

5.1.3.3 Numerical Implementation and Convergence Criteria

The product manifold optimization problem (eq. (99)) is solved using `pymanopt`'s implementation of the conjugate gradient (CG) algorithm for manifolds. Since the log marginal likelihood function is non-convex, gradient-based algorithms tend to get trapped in local minima. To decrease the chances of selecting a local optimum, this work uses n_{restart} restarts in the training procedure. Detection of local optima is handled by enforcing a heuristic that

stops the search if the relative improvement in the objective is less than $\epsilon_{\text{restart}}$ for n_{streak} iterations. After the n_{restart} exploratory CG searches, a final CG search is initialized from the best solution seen thus far. This final CG search is performed until either the step size between iterates or the norm of the gradient decreases below a threshold while a specified maximum number of iterations have not been exceeded. Algorithm 7 describes the numerical procedure for training the modified GP proposed in this work.

Algorithm 7: Training Algorithm for the Manifold Optimization

Input: $\boldsymbol{M} \in \mathbb{R}^{d \times n}$, $\boldsymbol{Y} \in \mathbb{R}^{n}$, q, $\epsilon_{\text{restart}}$, n_{streak}, $n_{\text{CG iter}}$
Output: $\tilde{\boldsymbol{\Xi}}$, $\boldsymbol{\theta}'$, σ_y

Initialize $-\mathcal{L}_{\text{best}} = \infty$, $\tilde{\boldsymbol{\Xi}}_{\text{best}} \in \mathbb{R}^{d \times q}$, $\boldsymbol{\theta}'_{\text{best}}$, $\sigma_{y,\text{ best}}$, $n_{\text{no improve}} = 0$
/* exploration; perform restarts */
for $i = 1, \ldots, n_{\text{restarts}}$ **do**
    $-\mathcal{L}_1 = \infty$
    for $j = 1, \ldots, n_{\text{CG iter}}$ **do**
        $-\mathcal{L}_{j+1}$, $\tilde{\boldsymbol{\Xi}}$, $\boldsymbol{\theta}'$, $\sigma_y = \texttt{pymanopt\_CG}(\tilde{\boldsymbol{\Xi}}, \boldsymbol{\theta}', \sigma_y, n_{\text{iter}} = 1)$ // return next iterate
        if $\frac{\mathcal{L}_{j+1} - \mathcal{L}_j}{\mathcal{L}_j} \leq \epsilon_{\text{restart}}$ **then**
            $n_{\text{no improve}} = n_{\text{no improve}} + 1$
        else
            $n_{\text{no improve}} = 0$
        if $n_{\text{no improve}} \geq n_{\text{streak}}$ **then**
            break
    if $\mathcal{L}_{j+1} > \mathcal{L}_{\text{best}}$ **then**
        $\mathcal{L}_{\text{best}} = \mathcal{L}_{j+1}$
        $\tilde{\boldsymbol{\Xi}}_{\text{best}}$, $\boldsymbol{\theta}'_{\text{best}}$, $\sigma_{y,\text{ best}} = \tilde{\boldsymbol{\Xi}}$, $\boldsymbol{\theta}'$, σ_y
/* Perform final search using best solution initialization */
$-\mathcal{L}_{\text{best}}$, $\tilde{\boldsymbol{\Xi}}_{\text{best}}$, $\boldsymbol{\theta}'_{\text{best}}$, $\sigma_{y,\text{ best}} = \texttt{pymanopt\_CG}(\tilde{\boldsymbol{\Xi}}_{\text{best}}, \boldsymbol{\theta}'_{\text{best}}, \sigma_{y,\text{ best}}, n_{\text{iter}} = n_{\text{CG iter}})$ // until convergence
Return $\tilde{\boldsymbol{\Xi}}_{\text{best}}$, $\boldsymbol{\theta}'_{\text{best}}$, $\sigma_{y,\text{ best}}$

5.1.4 Active Subspace using Optimization-Based Gaussian Process Regression

Another recently proposed method [265] this dissertation refers to as the Gaussian Process-Active Subspace (GP-AS) first builds a GPR model in the original high-dimensional input space. The novelty of their approach [265] lies in capitalization of the analytical derivation for the uncentered covariance matrix of the gradient for a GPR model. Once the model is trained, the method uses the analytical expression for the uncentered covariance of the

gradients and takes its SVD to compute the AS. Note that this approach is akin to the ASM in that the AS is computed by performing the SVD of the matrix $\boldsymbol{C}$ using its approximation obtained using a trained GP on the full-dimensional space rather than the gradient samples of the true function.

The authors use this technique as a means to define an acquisition function for performing adaptive sampling to improve the knowledge of the AS. Their experiments have shown promise on several test functions and therefore will serve as the benchmark method with which the MO-GP will be compared. Moreover, the AS-GP method is applicable under the same constraints/requirements regarding the dataset as the MO-GP method. But as we discussed before, GPR models tend to struggle in higher input dimensions owing to their dependence on distance-based kernels. Therefore, it is hoped that our approach performs at least as well, if not better because the MO-GP first projects and then trains the GP in a lower-dimensional space. This dissertation makes two key modifications to the implementation of this method to make the comparison fair. First, this dissertation opts for an MLE estimate for the GP hyperparameters instead of a fully Bayesian treatment. Second, instead of a fully analytical derivation of the uncentered covariance matrix, we adopt a semi-analytical approach that samples 1000 analytically evaluated gradients from the trained GP in the interest of computational efficiency. The analytical expressions used for the computation of the gradient of a GP are adopted from the derivation in [171].

5.2 POD and Interpolation-Based ROM with Input Space Dimension Reduction

5.2.1 Overall Concept

As mentioned before, the primary goal of this research area is the development of a method that leverages desirable properties of the POD-based ROM and discovery of a meaningful low-dimensional input subspace to simultaneously reduce the dimensionality of both the input and output spaces. The basic idea begins with the truncated POD ansatz restated here. A zero mean field is assumed without loss of generality for convenience in explanation.

$$\tilde{\boldsymbol{w}}(\boldsymbol{\mu}) = \boldsymbol{\Phi}_k \boldsymbol{h}(\boldsymbol{\mu})$$

where the parameter $\boldsymbol{\mu} \in \mathbb{R}^d$ is high-dimensional, i.e., $d \gg 1$. The goal is to capture the map $\boldsymbol{h} : \mathbb{R}^d \to \mathbb{R}^k$ from the parameters $\boldsymbol{\mu}$ to the latent space coordinates through *ridge approximations*. Moreover, both the input subspace and the approximating function are solved for simultaneously in a completely gradient-free manner using the novel dimension reduction kernel-based GP detailed in the previous section.

5.2.2 Manifold Optimization GPR-based Ridge Approximations in the POD Subspace

Two possibilities exist when considering the ridge approximation of a vector-valued function, say $\boldsymbol{h}$. If each component function is treated as a separate scalar, then one strategy is to build a different low-dimensional subspace for each component. Another possibility concerns the ability of a single active subspace to capture and characterize the majority of the variation in $\boldsymbol{h}$. It is neither obvious nor straightforward to comment about the relative effectiveness of either of the two aforementioned approaches for use in the latent space formed by the POD modes. Recent efforts [277] have successfully extended the ASM to vector-valued outputs where a common active subspace is constructed by minimizing an upper bound for the ridge approximation error of the vector-valued function.

However, construction of a surrogate model was not the end goal of the effort. In another effort [121], authors identified a shared subspace for vector-valued outputs by solving a least-squares problem that combines the active subspaces of multiple scalar-valued outputs. Although successful in practice, whether the combination is optimal or even interpretable is ambiguous.

For the ridge approximations and input subspace discovery of $\boldsymbol{h}$, the following strategy is proposed. Let the vector-valued latent space function h be represented as

$$\boldsymbol{h}(\boldsymbol{\mu}) = \begin{bmatrix} h_1(\boldsymbol{\mu}) & h_2(\boldsymbol{\mu}) & \dots & h_k(\boldsymbol{\mu}) \end{bmatrix}^T \tag{101}$$

where $h_i(\boldsymbol{\mu})$ is the i^{th} scalar component function of $\boldsymbol{h}$. The proposed strategy is to discover separate low-dimensional input subspaces corresponding to each $h_i(\boldsymbol{\mu})$ and construct ridge approximations given by

$$h_i(\boldsymbol{\mu}) \approx g_i(\boldsymbol{\Xi}_{q_i}^T \boldsymbol{\mu}) \quad \forall \quad i = 1, 2, \dots, k \tag{102}$$

where $\boldsymbol{\Xi}_{q_i} \in \mathbb{R}^{d \times q_i}$ is the reduced input subspace of dimension $q_i \leq d$. Note that this formulation accommodates subspaces of different sizes for every h_i. It has recently shown promise [110] in a setting where the gradients $\partial h_i / \partial \boldsymbol{\mu}$ are made available intrusively via an implementation of the adjoint method. In this work, the approximations g_i are trained in a gradient-free manner using the novel GP introduced in section 5.1.3. Prediction of the field at a new parameter with this strategy is made using

$$\tilde{\boldsymbol{w}}(\boldsymbol{\mu}) = \begin{bmatrix} \vdots & \vdots & \vdots \\ \boldsymbol{\phi}_1 & \dots & \boldsymbol{\phi}_k \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} g_1(\boldsymbol{\Xi}_{q_1}^T \boldsymbol{\mu}) \\ \vdots \\ g_k(\boldsymbol{\Xi}_{q_k}^T \boldsymbol{\mu}) \end{bmatrix}$$

The second strategy is inspired by the work in [277]. The latent space coordinates h are considered as the original vector-valued function to be approximated. This strategy utilizes an approximation of its Jacobian $\partial h / \partial \boldsymbol{\mu}$ and constructs an estimate of a common active subspace shared by all the components h_i of $\boldsymbol{h}$ resulting in the ridge approximations given by

$$h_i(\boldsymbol{\mu}) \approx g_i(\boldsymbol{\Xi}_q^T \boldsymbol{\mu}) \quad \forall \quad i = 1, 2, \dots, k \tag{103}$$

Note the difference between eq. (102) and eq. (103). The latter uses an identical common subspace to construct each ridge approximation. The resulting predictive ROM equation becomes

$$\tilde{\boldsymbol{w}}(\boldsymbol{\mu}) = \begin{bmatrix} \vdots & \vdots & \vdots \\ \boldsymbol{\phi}_1 & \dots & \boldsymbol{\phi}_k \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} g_1(\boldsymbol{\Xi}_q^T \boldsymbol{\mu}) \\ \vdots \\ g_k(\boldsymbol{\Xi}_q^T \boldsymbol{\mu}) \end{bmatrix}$$

This work pursues the first strategy as an initial step and defers the second strategy for future work. Algorithm 8 shows the steps involved in constructing POD and interpolation-based ROMs using the proposed approach.

Algorithm 8: Construction of the POD and Interpolation-Based ROM using the MO-GP model

Input:
Data: $\boldsymbol{M} \in \mathbb{R}^{d \times n}$, $\boldsymbol{W} \in \mathbb{R}^{m \times n}$
POD: k (POD truncation-rank)
GP: $\epsilon_{\text{restart}}$, n_{streak}, $n_{\text{CG iter}}$, $n_{\text{subspace dim.}} = \{q_i \forall i = 1, \ldots, k\}$
Output: $\bar{\boldsymbol{w}}$, $\{\ \tilde{\boldsymbol{\Xi}}_i \in \mathbb{R}^{d \times q_i}$, $\boldsymbol{\theta}'_i$, $\sigma_{y,\,i}\ \forall i = 1, \ldots, k\ \}$

Compute mean $\bar{\boldsymbol{w}} = \texttt{mean}(\boldsymbol{W})$
Subtract mean $\boldsymbol{W} = \boldsymbol{W} - \bar{\boldsymbol{w}}$
Compute truncated POD basis $\boldsymbol{\Phi}_k = \texttt{POD}(\boldsymbol{W},\ k)$
Project $\boldsymbol{z} = \boldsymbol{\Phi}_k^T \boldsymbol{W}$
/* train modified GP for each POD coefficient */
for $i = 1, \ldots, k$ **do**
    $\tilde{\boldsymbol{\Xi}}_i$, $\boldsymbol{\theta}'_i$, $\sigma_{y,\,i}$ = MO_GPR $(\boldsymbol{M},\ \boldsymbol{z}(:,\ i),\ q_i,\ \epsilon_{\text{restart}},\ n_{\text{streak}},\ n_{\text{CG iter}})$ // Estimate subspace and train GP using Alg. 7
Return $\bar{\boldsymbol{w}}$, $\{\ \tilde{\boldsymbol{\Xi}}_i \in \mathbb{R}^{d \times q_i}$, $\boldsymbol{\theta}'_i$, $\sigma_{y,\,i}\ \forall i = 1, \ldots, k\ \}$

5.3 Test Problem Description

The performance of the gradient-free manifold optimization-based approach for the input dimension reduction kernel-based GP is first evaluated using scalar functions. A thorough empirical evaluation of performance characteristics and comparison with another benchmark method is an absolute necessity before we proceed with recommendation of its use for constructing ROMs.

Following the testing on scalar functions, we use some field emulation problems to quantify the performance and demonstrate the benefits of the proposed method. First, a set of canonical elliptic PDE problems with different number of inputs are used to verify that the Manifold Optimization-based Gaussian Process (MO-GP) performs as expected for surrogates in the POD subspace.

Table 5: Summary of the Scalar Problems

Group	Function	d	q	Size
Synthetic	Quadratic	10	1	1000
	Quadratic	10	2	1000
	Quadratic	10	5	1000
	Quadratic	25	1	1000
	Quadratic	25	2	1000
	Quadratic	25	5	1000
	Quadratic	50	1	1000
	Quadratic	50	2	1000
	Quadratic	50	5	1000
	Quadratic	100	1	1000
	Quadratic	100	2	1000
	Quadratic	100	5	1000
Engineering Analytical	OTL Circuit	6	3	2000
	Borehole	8	4	2000
	Wing Weight	10	3	2000
	Power Circuit	13	1	2000
Engineering Datasets	NACA 0012 (Lift) [72]	18	1	1756
	HIV at $t = 3400$ [156]	27	1	1000
	ONERA M6 (Lift) [159]	50	1	297
	Elliptic PDE [76]	100	1	1000

5.3.1 Scalar Functions

The batch of scalar test problems contains some randomly generated quadratic functions and datasets from Constantine's AS repository[1]. The quadratic functions with known low-dimensional input subspace are given by

$$g(\boldsymbol{\Xi}^T\boldsymbol{\mu}) = \boldsymbol{\mu}^T\boldsymbol{\Xi}\boldsymbol{A}\boldsymbol{\Xi}^T\boldsymbol{\mu} + \boldsymbol{b}^T\boldsymbol{\Xi}^T\boldsymbol{\mu} + c + \epsilon \tag{104}$$

where $g : \mathbb{R}^q \to \mathbb{R}$ is the ridge function and $\epsilon \sim \mathcal{N}(0, 1)$ is standard Gaussian noise. The active subspace $\boldsymbol{\Xi} \in \mathbb{R}^{d\times q}$ and the function parameters $\boldsymbol{A} \in \mathbb{R}^{q\times q}$, $\boldsymbol{b} \in \mathbb{R}^q$ and $c \in \mathbb{R}$ are generated randomly. In order to make a thorough empirical assessment of the predictive performance of the proposed method, synthetic quadratic functions for several values of d and q are created as listed in table 5. Performance on candidate engineering problems is tested on various analytical functions and publicly available dataset (details in table 5). The gradient information for all the samples in the engineering functions and datasets is used to compute the true AS. Although the analytical gradients are available for the synthetic quadratic functions, the gradients at all the sampled points are used to estimate the true AS.

5.3.2 ROM Application Problems

This section introduces the four test problems used for the purpose of demonstration under this research area. The mix consists of a canonical elliptic PDE field emulation problem for which the number of parameters can be varied. Its field output's gradient with respect to the parameters is available. As such, this problem allows us to check whether the method performs as expected. Lastly, the three practical high-dimensional fluid flow problems are described.

5.3.2.1 High-Dimensional Elliptic PDE Problem

The method for the construction of ROMs by discovering a low-dimensional input space in the POD basis (see sec. 5.2) is first demonstrated on the prediction of the field from an

[1] https://github.com/paulcon/as-data-sets

elliptic PDE equation with a random spatially varying diffusion coefficient $\kappa(\boldsymbol{s})$ given by

$$-\nabla \cdot \kappa(\boldsymbol{s}) \nabla w(\boldsymbol{s}) = 1, \quad \forall w \in [0,1]^2$$

Homogeneous boundary conditions are applied on the top, left, and bottom boundaries and a Neumann condition is enforced on the right boundary. The input parameters are coefficients of the following truncated KL expansion related to the κ,

$$\log \kappa(\boldsymbol{s}, \boldsymbol{\mu}) = \sum_{i=1}^{d} \mu_i \gamma_i \phi_i(\boldsymbol{s})$$

where ϕ_i and γ_i are the eigenpairs of the correlation function $\mathcal{C}(\boldsymbol{s}_1, \boldsymbol{s}_2) = \exp\left(-\beta^{-1} ||\boldsymbol{s}_1 - \boldsymbol{s}_2||_1\right)$, each $\mu_i \sim \mathcal{N}(0,1)$ and β is the correlation length. Smaller values of β lead to physically richer fields, i.e. a slower decay in the spectrum of the correlation function. In this study, $\beta = 5$ is used throughout. The spatial domain is discretized such that 10,000 unknowns are solved for, resulting in a snapshot size $m = 10,000$. For a thorough empirical assessment of the performance of the proposed method, ROMs are constructed for the systems resulting from KL expansions truncated at $d = 25,\ 50$ and 100 modes respectively.

In addition to 1,000 total samples for each problem, the adjoint-based gradient implementation is used to compute the true gradients of the entire field with respect to the input parameters. The subspace discovered through the gradient-free method is compared against

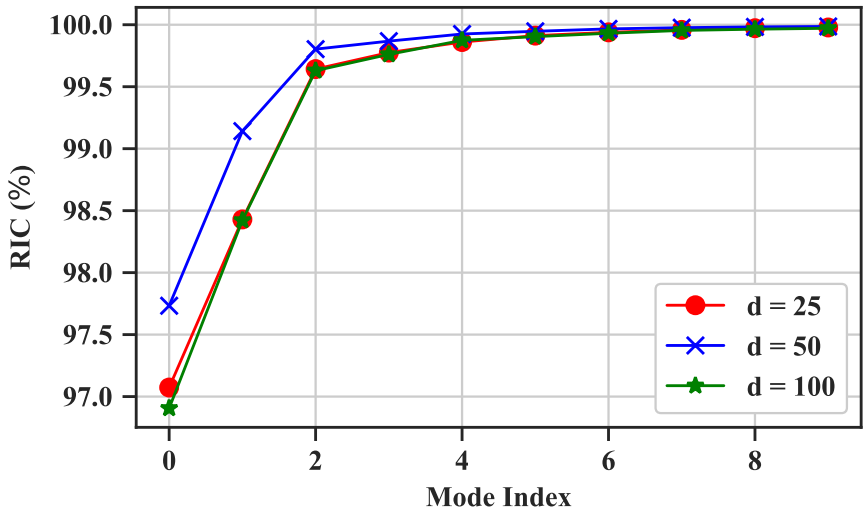


Figure 70: RIC for all the Elliptic PDE Field Prediction Problems

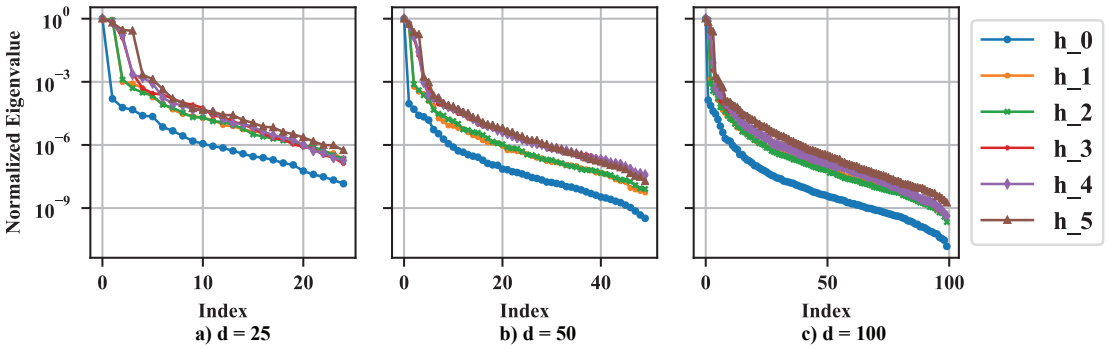


Figure 71: Spectrum of an Approximation to $\boldsymbol{C}$ for each POD coefficient

the AS computed using the approximate gradients for the POD latent space coordinates obtained by the projection of the true gradient onto the POD basis. Let $\boldsymbol{\Phi} \in \mathbb{R}^{m \times k}$ be the POD basis for a set of n snapshots of the full order system and let $J \in \mathbb{R}^{m \times d}$ be the Jacobian of the field with each row containing the gradient of the field solution at a spatial location with respect to the input parameters. Under the POD assumption, the Jacobian for the coordinates in the POD basis is given by

$$\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\mu}} = \boldsymbol{\Phi}_k^T \boldsymbol{J}, \quad \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\mu}} \in \mathbb{R}^{k \times d} \tag{105}$$

For all the three problem sizes across all the training sets, an RIC of 99.9% resulted in a rank-6 (see figure 70) truncation of the POD basis. The true AS dimension for each of the 6 POD coefficients was computed by applying eq. (105) to all the 1000 samples of the Jacobian. Figure 71 shows the decay of the eigenvalues of an approximation of $\boldsymbol{C}$ for each coefficient computed using the SVD on 1,000 gradient samples for each of the three problem sizes considered. The choice of the true AS dimension is determined by a large gap in the spectrum [72] which is clearly visible for all the POD coefficients. The true dimension of the AS ($q_i,\ i = 0, \ldots, 5$) is chosen such that the ratio between the first and the q_i^{th} eigenvalue exceeds 10. Table 6 shows the resulting true AS dimensions for each of the problem sizes. Note that the number of AS dimensions corresponding to the higher modes either increases or at most stays constant. This is perhaps because higher modes capture the nuanced features of the field. Therefore, their variation is captured by a relatively higher-dimensional input subspaces.

Table 6: Field Prediction Elliptic PDE Problem Details

Problem	Dim. Inputs (d)	Dim. AS (q_i)						Total Sample Size
		h_0	h_1	h_2	h_3	h_4	h_5	
1	25	1	2	2	2	3	4	1000
2	50	1	2	2	3	3	4	1000
3	100	1	2	2	3	3	4	1000

5.3.2.2 Subsonic Flow over Airfoils

Emulation of the pressure coefficient distribution around RAE 2822 airfoils with 2 different sets of parametrizations is chosen as the first demonstration task to test the developed methodology for constructing ROMs with high-dimensional inputs and outputs. The steady-state flow is obtained using an Euler implicit scheme for the RANS equations with the SA turbulence model using the SU2 code. The airfoil simulation setup for this test problem differs from the setup for research area 1 (see section 4.4.1) in the following ways: 1) the flow is simulated at a subsonic Mach number of 0.3, 2) the Free-Form Deformation (FFD) parametrization is relatively higher-dimensional, and 3) a coarser structured o-grid is used to discretize the spatial domain and the geometry.

The airfoils are parametrized using a relatively higher number of variables to make

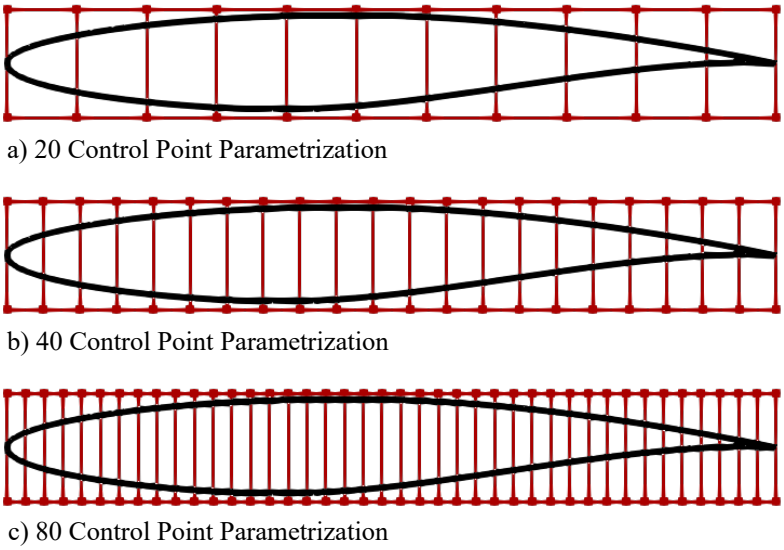


Figure 72: Examples of Parametrizations with Varying Number of FFD Control Points

them suitable candidates for this research area. In order to flexibly set the number of input parameters, the number of chord-wise FFD control points are parametrically varied as shown in figure 72. Instead of moving the points at a specific chord-wise location in the y-direction to change the camber of the airfoil (see section 4.4.1), we displace the points corresponding to the top and the bottom surfaces independently in the y-direction. This slight modification results in a richer set of shapes with varying thicknesses and camber. As a consequence, the richness in the features of the resulting flow fields increases. The coordinates of the control points at the leading and trailing edge are kept fixed. For our test cases, we use two sets of parametrizations as listed in the table 7. The first parameter for both the sets is the angle of attack (AoA). The subsequent parameters are the control points of the FFD boxes with 14 and 24 chord-wise stations (see examples of parametrizations with different numbers of chord-wise stations in fig. 72), respectively.

The choice to simulate a subsonic flow instead of the physically more challenging transonic flow was deliberate. The high-dimensional parametrization increases the richness in the physical features of the flow due to the larger degrees of freedom in the generation of airfoil shapes. Since the goal was to check whether large parameter spaces can be effectively handled by the proposed method, we did not want the transonic flow condition to undesirably interfere with the low-rank structure of the coefficient of pressure and unnecessarily contribute to the challenge.

Table 7: Details of the Parametrization for the Subsonic Airfoil Problems

Description	d	Lower Bound	Upper Bound	Total Sample Size
AoA and 14 FFD points	15	AoA: 0 deg. FFD: -0.03	4 deg. 0.03	2000
AoA and 24 FFD points	25	AoA: 0 deg. FFD: -0.03	4 deg. 0.03	2000

All the models for the airfoil problems predict the coefficient of pressure on a structured o-grid consisting of approximately 11,000 cells. Note that the number of cells is significantly

smaller (0.25x) than the transonic case in the problem in section 4.4.1. However, it is still high-dimensional for the purposes of this particular study. Each database of 2,000 field solutions is generated using a DoE containing designs generated using a pseudo-random Sobol sequence. The snapshot matrices thus obtained are of size $11,000 \times 2,000$.

As represented in figure 73, for an RIC of 99.9%, the SVD of the entire dataset of snapshots yields a truncation rank k of 13 and 15 for the 15 and 25 variable problems, respectively. The increase in the truncation rank due to the addition of design variables is indicative of an increase in the complexity of the flow field due to the more nuanced and flexible airfoil shapes.

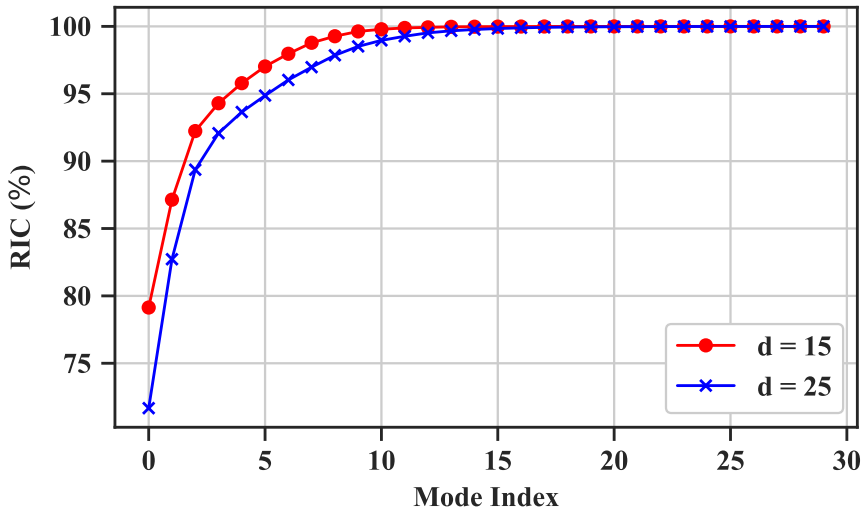


Figure 73: RIC for the High-Dimensional Airfoil Problems

5.3.2.3 Subsonic Flow over the CRM Wing

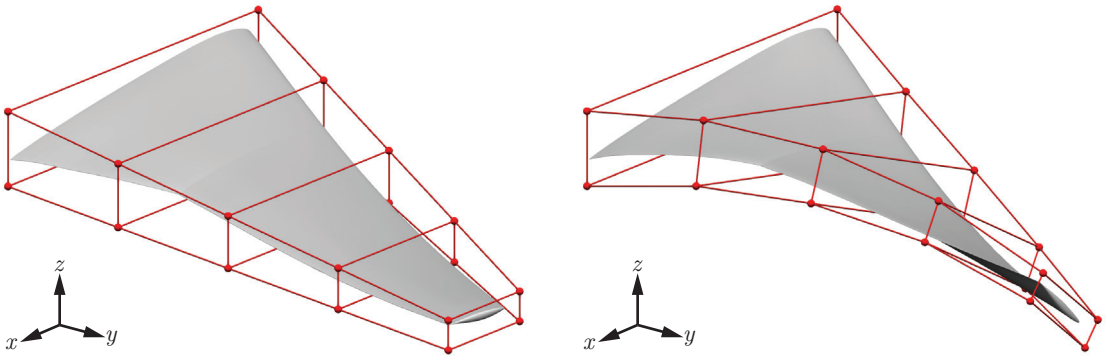
The geometry for the final practical problem under this research area is that of the CRM wing with a high-dimensional shape parametrization. The ROMs are built to emulate the 3-dimensional distribution of the pressure coefficient in subsonic flow conditions under varying wing shapes based on the FFD. The CFD simulation setup for the CRM wing closely resembles that of the airfoil described in the previous section. For the discretization of the fluid domain, we use the grid used for the CRM demonstration test case in chapter 4.

The flow field is simulated for an AoA of 2 deg. and Reynolds number of 5×10^6 at

a subsonic Mach number of 0.3. Note that the reasons for demonstration on a wing in subsonic flow are similar to the reasons discussed previously for the airfoil case. In fact, the increase in complexities due to a large number of design variables on an already complex transonic flow is more pronounced for the CRM wing because it is a three-dimensional geometry.

The high-dimensional shape parametrization is obtained by displacing the control points of the enclosing FFD box. The granularity of the parametrization can be flexibly increased/decreased by varying the number of control points on the FFD box, and by specifying the number or groups of points that are allowed to move. Figure 74 shows an example where groups of points along span-wise stations are rotated simultaneously to produce a twist in the wing.

For this specific example, the five span-wise stations result in a 5-dimensional parametrization. In our test case, instead of rotating groups of points, we independently displace each control point in the y-direction. We consider the parametrization from an FFD box with 4 span-wise and 3 chord-wise stations resulting in a total of 24 shape parameters that can each move independently in the y-direction. Figure 75 shows the FFD box along with the accompanying control points for our parametrization. The control points are allowed to move $+/-2.5\%$ in the y-direction from their respective baseline positions.



a) Original Geometry b) Deformed Geometry

Figure 74: Example of a Deformation of the CRM Wing using the FFD Parametrization. Exaggerated for Visual Appeal. Reproduced with Permission from [194].

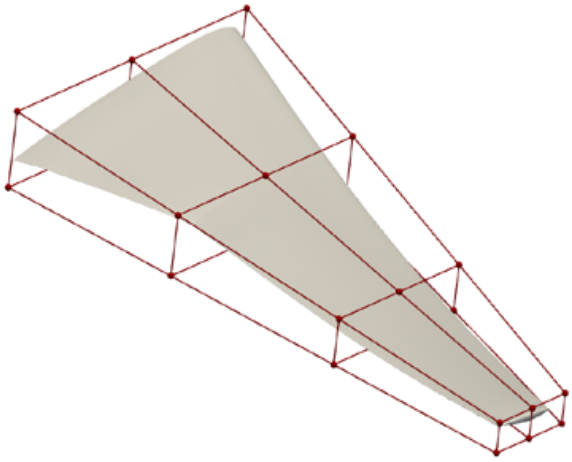


Figure 75: FFD Box with a 24 Control Point Parametrization

While the volume grid contains $450,000$ points, the ROMs for this problem predict the coefficient of pressure over the surface of the wing which consists of approximately $11,000$ discrete elements. The full dataset consists of $1,000$ field solutions obtained for wing shapes generated from a pseudo-random Sobol sequence leading to snapshot matrix of size $11,000 \times 1,000$. As shown in figure 76, an RIC of 99.9% results in a truncation rank k of 29 using the SVD algorithm on the entire dataset. Therefore, all the ROMs constructed for this problem use a k of 29.

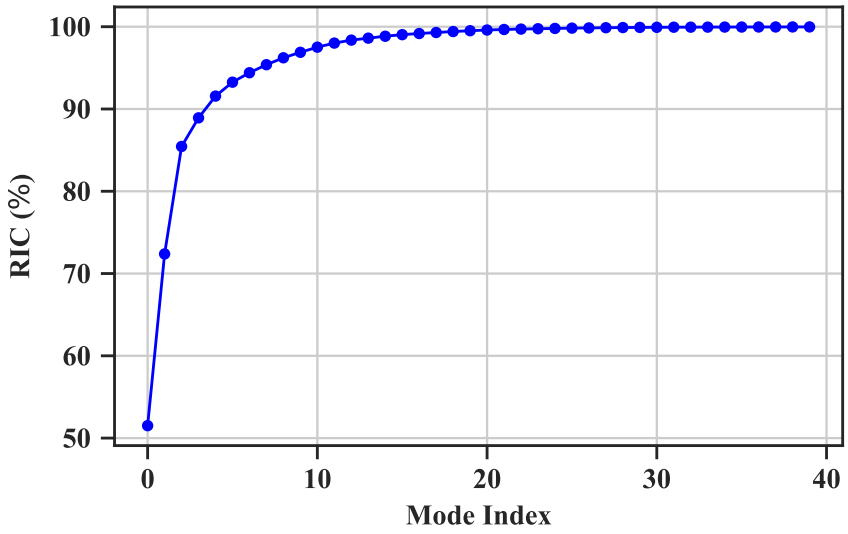


Figure 76: RIC for the High-Dimensional CRM Wing Problem

5.4 Experiments and Results

This section builds arguments for development of hypotheses for all the questions posed in chapter 3 under this research area, presents the questions along with the hypotheses, develops experiments to validate the hypotheses, and finally presents the results.

5.4.1 Subspaces Computed by Manifold Optimization-Based Gaussian Process Regression

We are now ready to start evaluating the characteristics of the MO-GP supervised learning model on scalar problems. To do so in a principled manner, the first question regarding the subspace found by the MO-GP was posed as:

Research Question 2.1

Can the proposed method successfully recover a meaningful low-dimensional input subspace, i.e., one that captures a majority of the true function's variation?

The argument for the hypothesis to this question lies in the definition of its MLE training procedure. The supervised learning model that was developed in section 5.1.3 finds a subspace (or a unique point belonging the Grassmann manifold) given its input space dimensionality and trains a GPR model that maximizes the log-marginal likelihood of observing the data in the subspace it finds. It is therefore reasonable to expect that the algorithm finds a meaningful subspace. In hypothesis form, it may be stated as follows:

Hypothesis 2.1: *Given the number of dimensions q of the low-dimensional input subspace, the training procedure for the MO-GP maximizes the log marginal likelihood by definition. If the optimization algorithm converges to an optimum that maximizes the log marginal likelihood of observing the data, then among all the possible subspaces of dimension q, the MO-GP yields a q-dimensional subspace which captures a majority of the function's variation.*

Recall that the ASM clearly defines the subspace in which the function varies more on average than others using gradient information. The MO-GP model on the other hand,

purely relies on the input-output pairs and the supervised learning procedure to find the low-dimensional subspace in which the GPR model maximizes the probability of observing the data. As a first empirical check for whether or not the subspace MO-GP finds is sensible, this experiment utilizes some of the sample test functions with a known input subspace dimension chosen using the heuristic for the ASM. The problem set contains an assortment of synthetic quadratic functions, analytical engineering functions, and real engineering datasets. The idea is to slowly increase the input space dimensionality and observe the predictive performance of the resulting surrogate model(s) for several training repetitions on different randomly sampled sets of training data. In order to prevent over-fitting and make sure that the optimization problem does not struggle with convergence due to lack of data, the training sizes are set to fairly large values for this experiment. If the resulting models perform consistently across all training repetitions and their performance increases at least until the input subspace dimension equals the AS dimension size, we can say that the subspace MO-GP yields is not arbitrary. In other words, the MO-GP finds a subspace that is optimal in the sense that the GP trained in it is consistently able to explain the variation in the training data accurately.

For all the surrogate building tasks on scalar functions, this dissertation uses the Normalized Root-Mean-Squared Error (NRMSE) metric to measure the deterministic predictive capability of the model. Let $y_{\text{i,predicted}}$ denote the i^{th} prediction made by the model in a set of n prediction points, the Root-Mean-Squared Error (RMSE) divided by the standard deviation in the n actual samples $\{y_{\text{i,actual}},\ i = 1, \ldots, n\}$ is called the NRMSE. It is written as

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} \left(y_{\text{i,predicted}} - y_{\text{i,actual}}\right)^2}{n}} \tag{106}$$

$$\sigma = \sqrt{\frac{\sum \left(y_{\text{i,actual}} - \mu\right)^2}{n}} \tag{107}$$

$$\text{NRMSE} = \frac{\text{RMSE}}{\sigma} \tag{108}$$

where μ is the sample mean. Furthermore, form here on the values of NRMSE computed for the training and validation sets are referred to as T-NRMSE and V-NRMSE respectively.

The normalization gives an intuitive feel to the error metric because of the scaling by

the natural variations in the data, i.e. it expresses the NRMSE as a fraction of the total standard deviation in the prediction set.

Details of the Experiment. The experiment trains MO-GP models beginning with a 1-dimensional input subspace for each function and keeps increasing it until it is slightly above the respective function's known AS dimension. For each function, we perform 20 training repetitions on randomly sampled training sets with 450 points each. The remaining points automatically assume the role of validation/test points. Both the training and validation NRMSE values are recorded and finally visualized. The number of restarts for the MO-GP algorithm was set to 500. Figure 77 summarizes the steps for this experiment.

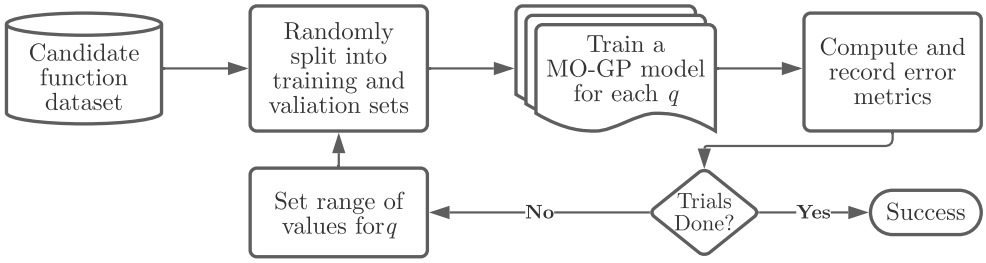


Figure 77: Setup for Experiment 2.1.

Results. Let us first discuss the results for the synthetic quadratic functions with 25 inputs as shown in figures 78 and 79. Note that these functions have been explicitly constructed to be 1- and 2-dimensional respectively (see eq. (104)). Observe that as the number of input dimensions increases for both quadratic functions, the accuracy with which the GP-AS model fits the training data increases as seen in figure 78. As expected, the MO-GP struggles with a 1D subspace for the 2-dimensional quadratic function as indicated by the relatively larger T-NRMSE values. As soon as the number of dimensions is increased to 2, the training errors reduce drastically. Additionally, the models achieve similar T-NRMSE values across all the training repetitions for a given input subspace dimension. However, beyond the exact size of the actual lower-dimensional subspace for the respective quadratic functions, there is a marked drop in the V-NRMSE values (see fig. 79). This behavior is indicative of the fact that adding more dimensions than required to the input space leads

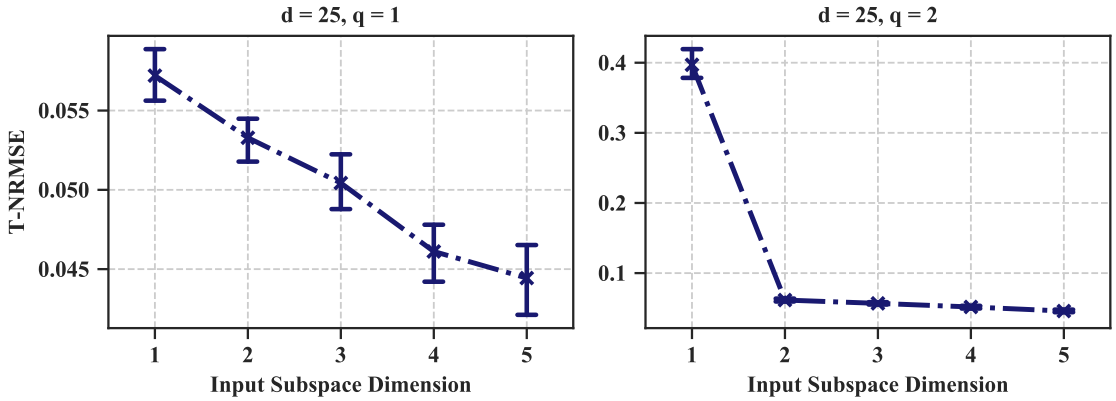


Figure 78: Variation of Training Normalized Root Mean Squared Error with Number of Input Dimensions for the Quadratic Problems

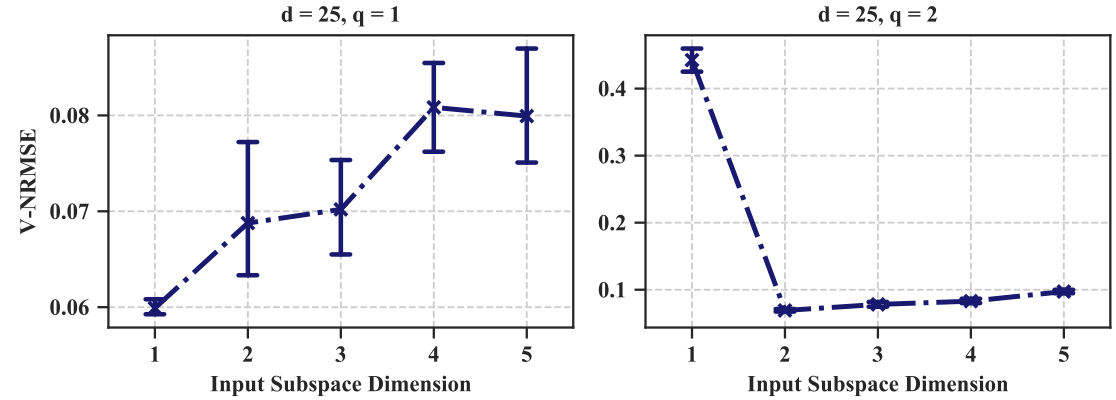


Figure 79: Variation of Validation Normalized Root Mean Squared Error with Number of Input Dimensions for the Quadratic Problems

to the model over-fitting to the noise introduced in the training set due to the projection. In an attempt to capture the noise using the additional dimensions, the computed subspace loses some accuracy on predicting the points in the validation set.

Figures 80 and 81 show the results for this experiment on engineering functions and datasets. Unlike the case of quadratics where the functions were constructed to be low-dimensional, the value of the low-dimensional input subspace q as suggested by the selection heuristic for the ASM is a subjective choice after all. Therefore, the V-NRMSE along with the T-NRMSE keeps decreasing as more dimensions are added for all the engineering functions, except the elliptic PDE. In the case of the quadratics, we introduced noise in the

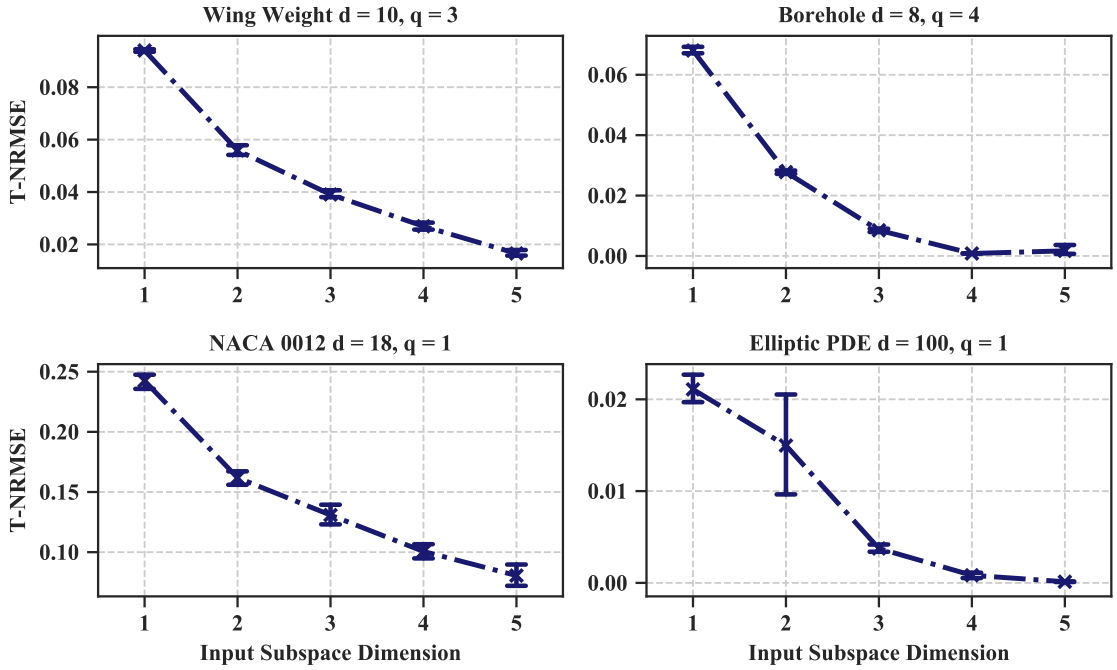


Figure 80: Variation of Training Normalized Root Mean Squared Error with Number of Input Dimensions for the Engineering Functions

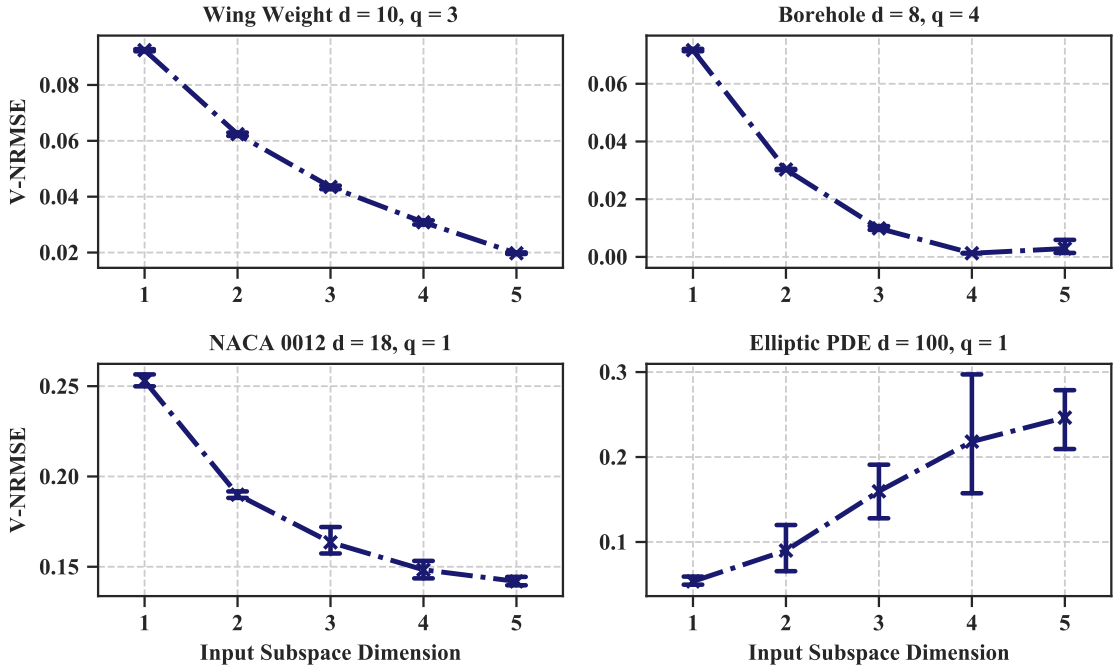


Figure 81: Variation of Validation Normalized Root Mean Squared Error with Number of Input Dimensions for the Engineering Functions

low-dimensional input subspace deliberately to demonstrate use with the MO-GP method. As such, the poor behavior of the V-NRMSE beyond the true input subspace dimension was distinct. However, in the case of engineering functions and datasets, having additional subspace dimensions successively helps capture information that is otherwise forced to appear as noise in lower-dimensional subspaces. Note that similar to the results for the quadratic functions, the T- and V-NRMSE values are consistent across 20 training repetitions with randomly sampled training datasets of sufficient size.

These experiments have shown that the MO-GP model either finds a meaningful subspace or finds a good GP model in some low-dimensional subspace. In other words, whether the subspace found by the MO-GP resembles the known Active Subspace is unclear. Nevertheless, the combination of the GP model and the manifold provides an optimal low-dimensional subspace in the sense that the GPR performance is better in that subspace than any other. Moreover, MO-GP achieves this performance consistently when provided with a sufficiently large (enough for finding the same subspace over all the repetitions consistently) number of training samples.

Summary. The findings of this experiment have shown that the MO-GP method is successful in its primary goal of consistently finding a low-dimensional subspace and an accompanying model in that subspace such that the probability of observing the data is maximized. The behavior of the predictive accuracy metrics indicates that when given the true dimensionality of an exactly low-dimensional function, the MO-GP method gives the best model when its value for q is set to the correct value. For engineering functions and datasets, on the other hand, the MO-GP seems to capture the noise introduced due to projection when given any number of dimensions beyond the AS dimension.

5.4.2 Empirical Assessment of the MO-GP Model

Following a quick preliminary experiment to verify that the MO-GP model computes a meaningful subspace and accompanying model, we pose the following question to perform empirical tests and understand the subspace found by the MO-GP model and its relationship to the AS in a deeper manner. The question also places the MO-GP in terms of its

performance by comparing it to the contemporary GP-AS method with respect to several considerations.

Research Question 2.2

Does the subspace recovered by the proposed model have any relationship to the Active Subspace? How competitive is the model's capacity to identify the subspace and yield good generalizability with respect to:

1. the number of inputs
2. the number of training samples
3. the size of the actual low-dimensional input subspace?

Recall that the supervised learning model that was developed in section 5.1.3 finds a point in the Grassmann manifold given its input space dimensionality and trains a GPR model that maximizes the log marginal likelihood of observing the data in the subspace it finds. It would not be surprising if the algorithm actually finds a subspace similar to the AS. In fact, it appeals to intuition that the subspace MO-GP finds, given sufficient training data, is indeed the AS. However, there exists no precise mathematical argument to base this intuition on. Fortunately, if the goal is to discover some structure in the inputs and obtain a good predictive performance, the relationship between the AS and MO-GPs low-dimensional subspace is inconsequential. Moreover, due to our deliberate formulation of the MO-GP in the Grassmann manifold, the low-dimensional subspace shares with the subspace computed by the ASM, the desirable property of orthogonality. In hypothesis form, our expectation can be stated as:

Hypothesis 2.2: *Since there exists no obvious relationship between the gradient-free MO-GP and the ASM, the subspaces need not necessarily be related. The AS-GP method fundamentally depends on the GP's accuracy to effectively find the AS. But GPs are known to struggle in high-dimensions. Therefore, the MO-GP will at least be competitive if not superior to the AS-GP method with respect to all the considerations listed above. If the MO-GP*

method is competitive in terms of its predictive accuracy in comparison to the benchmark GP-AS method, whether or not the subspaces computed by them are related bears no consequence.

To quantify the relationship between the AS and the subspace found by the MO-GP, we use the principal subspace angles metric introduced in chapter 4. The cosines of the angles between subspaces corresponding to $\boldsymbol{\Phi}_{k1} \in \mathbb{R}^{m \times k_1}$ and $\boldsymbol{\Phi}_{k2} \in \mathbb{R}^{m \times k_2}$ is given by the singular values of

$$\boldsymbol{S} = \boldsymbol{\Phi}_{k1}^T \boldsymbol{\Phi}_{k2}, \quad \boldsymbol{S} \in \mathbb{R}^{k1 \times k2}$$

In practice, the sine is preferred for computing the angles instead of the cosine due to numerical issues. The smaller the discrepancy in the angles between the true AS and the subspace computed by the MO-GP, the closer is its ability to recover the AS. We only report the largest principal subspace angle (shortly denoted as LPSA) because it represents an upper bound on the discrepancy between subspaces. Reporting one angle instead of many also facilitates easy comparison. For all the datasets listed in Table 5, the actual AS is computed using the original ASM detailed in section 5.1.1. For the engineering datasets/functions that do not have a published value of the size of the true AS, we set it equal to the index at which the spectrum of the Monte-Carlo approximation of $\boldsymbol{C}$ has decreased in value by over an order of magnitude. In order to minimize the loss in quality of the estimate of $\boldsymbol{C} = \sum_{i=0}^{n} \nabla f(\boldsymbol{\mu}_i) \nabla f(\boldsymbol{\mu}_i)^T$, we use the entire dataset to compute the true AS.

The deterministic predictive capability of the models is compared using the V-NRMSE (eq. (106)). This metric allows comparisons across datasets and models because of the normalization by the standard deviation of the prediction set.

Details of the Experiment. The comparison between the GP-AS and MO-GP methods is performed on all the scalar functions described in Table 5. To ensure a fair and thorough comparison, the mix of functions is comprised of a wide range of input dimensions (denoted by d) and AS dimensions (denoted by q). While the quadratic functions allow for the easy

modification of both d and q but represent synthetic functions, the analytical functions and datasets represent situations routinely encountered in practice. Note that the variety of functions considered covers performance comparisons with respect to the number of inputs and the size of the actual low-dimensional subspace.

To benchmark the methods with respect to the number of training samples (denoted by n) required to comfortably discover the subspace, this experiment constructs models for a range of training samples expressed in multiples of the respective function's input space dimensionality d. For each function category, GP-AS and MO-GP models are constructed with a number of training samples varying from 1x-8x the number of input dimensions d. For the MO-GP models, the dimensionality of the low-dimensional subspace is fixed to the size q of the AS. The LPSA is computed using the true AS for each function obtained using the entire dataset. The MO-GP method directly computes the input projection subspace. For the GP-AS method, 1000 gradient samples are drawn to compute the AS. The results are reported for 20 repetitions trained with randomly sampled training sets. The points not included in the training process constitute the validation set. The set they form is used to compute the V-NRMSE error metric. The MO-GP method is trained using $n_{\text{restarts}} = 250$. In spite of consistent performance with 100 restarts, the number was deliberately increased for the purposes of this experiment to ensure that the optimizer had a sufficient number of chances to find a good optimum. Each restart was terminated if the objective did not improve by 0.1 % for 50 consecutive CG steps. Figure 82 summarizes the steps for this experiment.

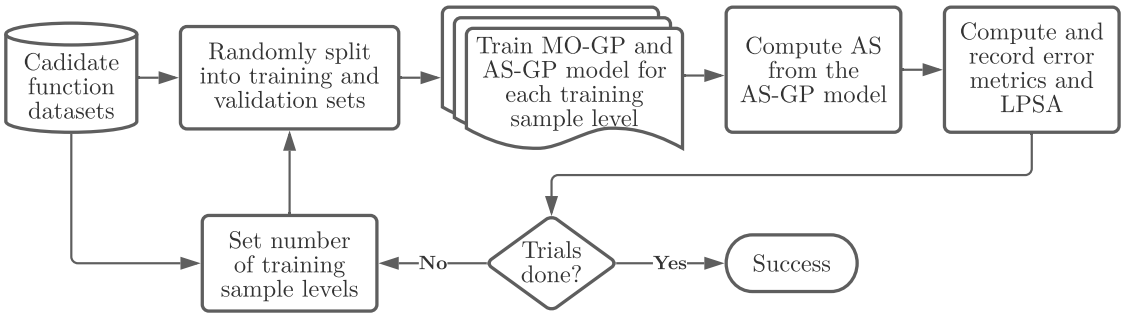


Figure 82: Setup for Experiment 2.2.

Results for the Quadratic Functions. Figure 83 and Figure 84 show how the LPSA between the true AS and the subspace computed by the MO-GP and GP-AS varies with the number of training samples. Recall that an LPSA of 0 indicates that the subspaces are aligned i.e. identical. We report angles in radians.

It is clear from figure 83 and figure 84 that the MO-GP method does find the true AS without using gradients. Despite the absence of a precise reason for the subspaces being similar, it can be observed that just the MLE optimization procedure finds the AS. The variability in the LPSA occurs due to randomness in the training set across the training repetitions. Consequently, a larger spread is observed with small training sets because of the higher chances of picking samples belonging to disparate localized regions in the domain. A first cursory glance reveals that for all the quadratic problems, the number of training samples at which the true AS is recovered (indicated by a sharp drop in the LPSA) by both the GP-AS and the MO-GP depends on the number of inputs and the size of the AS.

The MO-GP performs as well as the GP-AS method for 1x-2x the number of training samples. Beyond $n = 2 \times d$, the MO-GP consistently outperforms the GP-AS in its ability to recover the true AS across all the quadratic problems. For quadratic functions with small input dimensionality $d = 10$, this indicates the superiority of the MO-GP method over the GP-AS method because the GP is expected to be effective in 10 dimensions. Beyond $d = 25$, the onset of the *curse of dimensionality* negatively affects the GP-AS method because it first trains the surrogate model in the full input space and then computes the AS. In general, note that the MO-GP requires a larger number of training samples to discover the true AS at larger values of q across all the quadratic problems with $d = 10, 25, 50$ and 100.

Finally, note a little more nuanced observation regarding the effect of d for a given q on the number of training samples required. Across all the quadratic functions for a given q (read fig. 83 and fig. 84 column-wise), the MO-GP requires a larger number of training samples to find the true AS as d increases. A possible explanation for this behavior is the larger and harder optimization problem that the MO-GP requires solving on product manifolds.

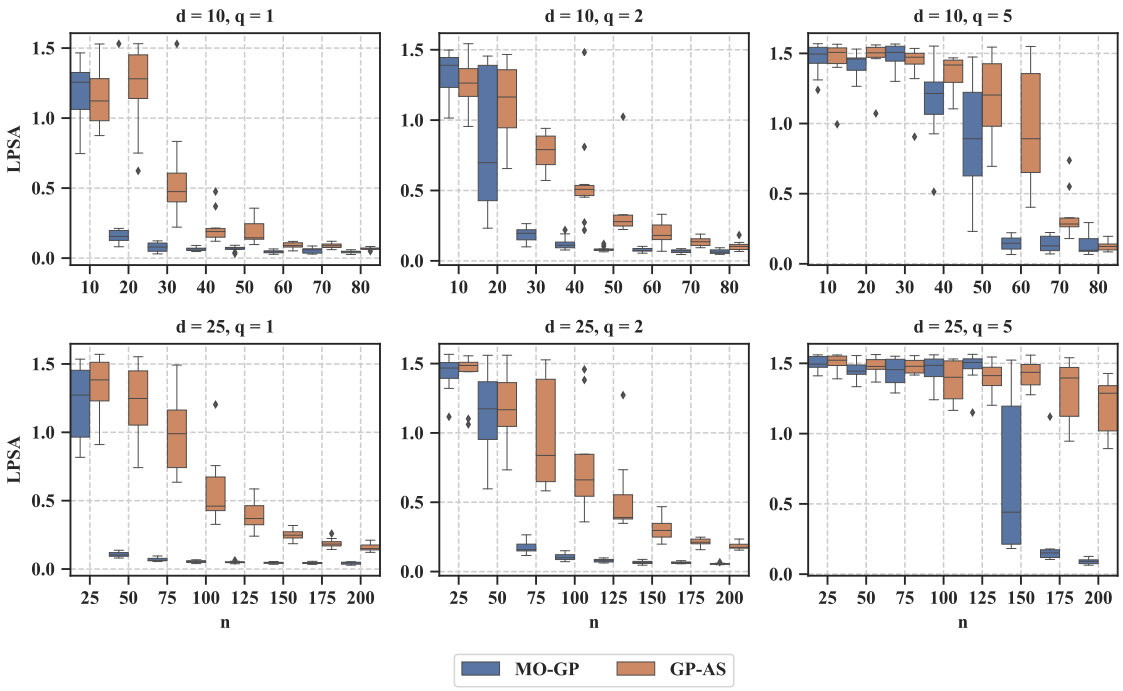


Figure 83: Variation of Largest Subspace Angle with Number of Input Dimensions for the Quadratic Problems with Input Dimensions 10 and 25

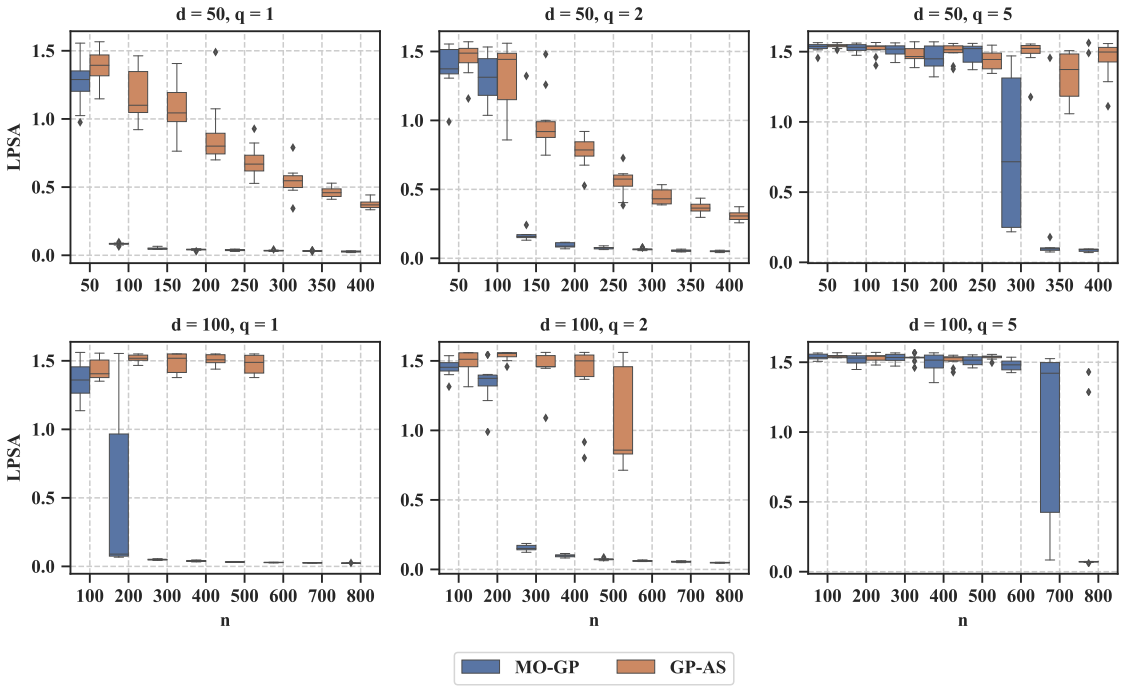


Figure 84: Variation of Largest Subspace Angle with Number of Input Dimensions for the Quadratic Problems with Input Dimensions 50 and 100

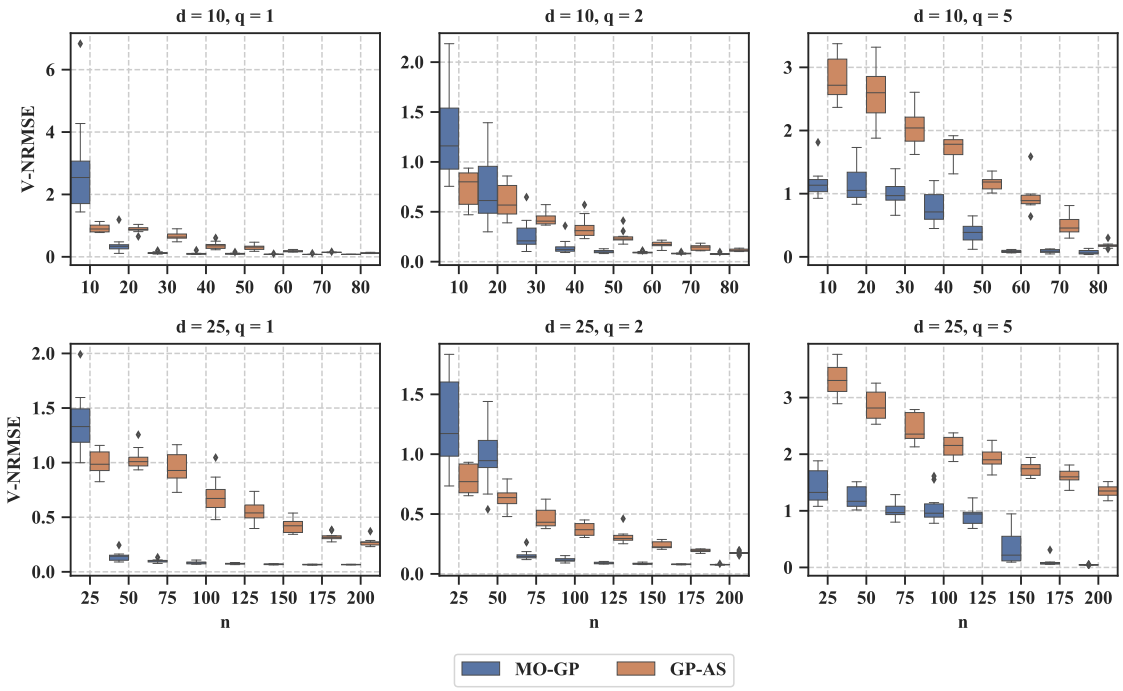


Figure 85: Variation of Validation Normalized Root Mean Squared Error with Number of Training Samples for the Quadratic Problems with Input Dimensions 10 and 25

The predictive capabilities of the GP-AS and MO-GP methods as measured by the NRMSE for quadratic functions are shown in figure 85 and figure 86. The trends are consistent with the observations made for the LPSA. In particular, the MO-GP is competitive or slightly worse off when the number of training samples is 1x-2x the number of input dimensions. However, beyond $n = 2$ or $3 \times d$ the MO-GP method performs significantly better than the GP-AS method. While the onset of the sharp drop in the NRMSE approximately coincides with that of the LPSA for all the quadratic problems, notice that in general, the NRMSE starts taking values less than 1 before the LPSA approaches 0. This behavior can be explained by recalling the definition of the LPSA. The LPSA only indicates the worst angle between the AS and the subspace computed by the MO-GP. However, it is possible for the MO-GP to partially discover the AS, i.e. the discrepancy in the LPSA is exclusively due to the relatively insignificant trailing dimensions. On further inspection, we found that this is in fact the reason for the relatively sooner decline in the NRMSE. Because the quadratic functions are precisely designed to be low dimensional, this behavior is rather weak.

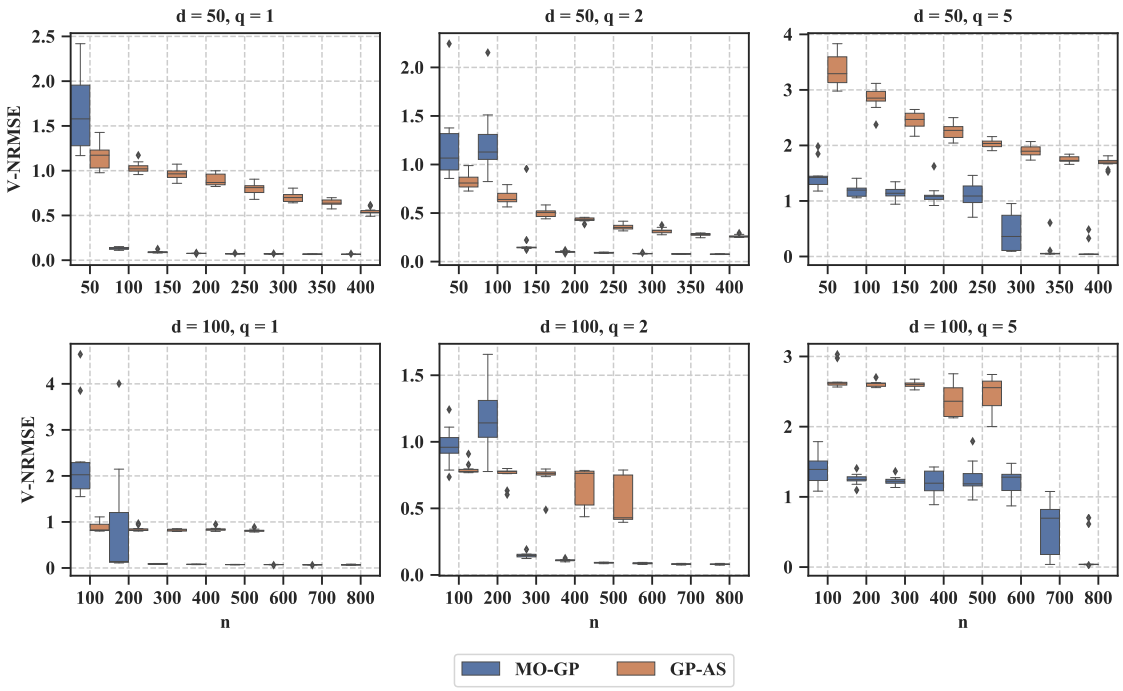


Figure 86: Variation of Validation Normalized Root Mean Squared Error with Number of Training Samples for the Quadratic Problems with Input Dimensions 50 and 100

Results for the Engineering Functions and Datasets. We now make observations for engineering functions and datasets similar in nature to the ones made for quadratic functions. Before proceeding, recall that unlike the precisely low-dimensional quadratic functions, the engineering functions and datasets do not have an innate low-dimensional structure. The size q of the true AS is determined by truncating the spectrum of $\boldsymbol{C}$ such that the eigenvalues have dropped by a specified order of magnitude. Figure 87 clearly shows that even for functions with relatively small $d \approx 10$, the MO-GP is superior in its ability to find the true AS with small training sizes. For larger training sample sizes $n \geq 4 \times d$ however, the GP-AS catches up as expected and starts capturing the AS with higher accuracy than the MO-GP. Since the GP-AS method actually computes the AS a posteriori, its accuracy depends on the quality of the GP model in the full-dimensional space. Therefore, when sufficient training samples are available, it starts outperforming the MO-GP method. On the other hand, the MO-GP method is asked to find the subspace given its size q. The GP in the projected q-dimensional space performs better with fewer training samples.

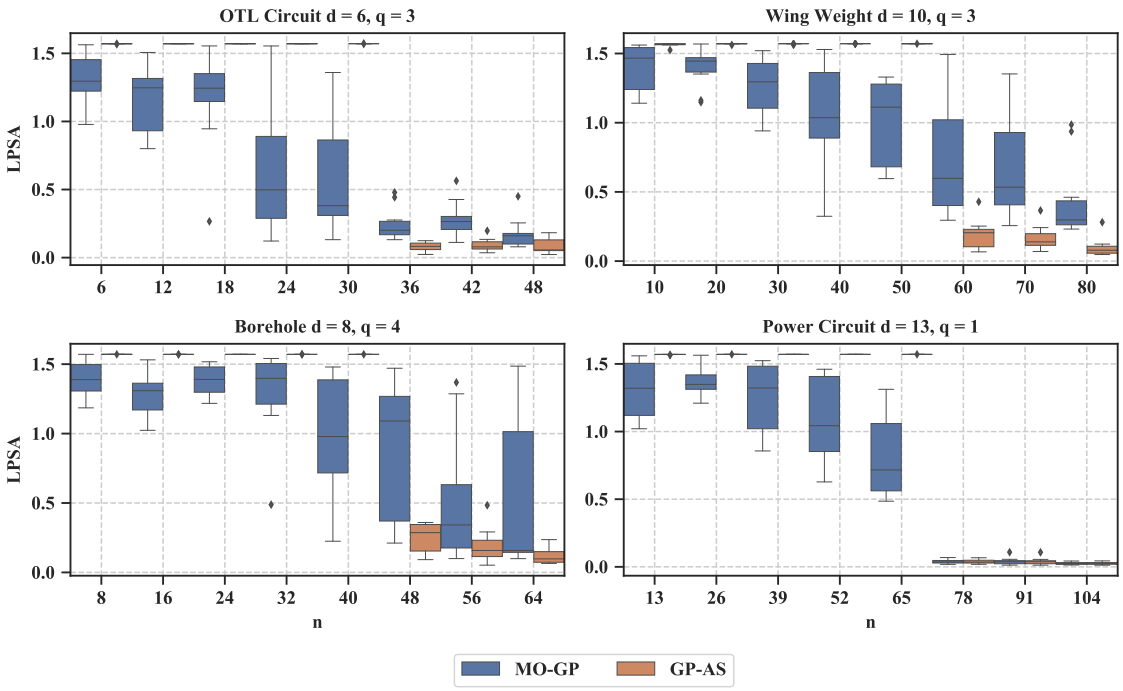


Figure 87: Variation of Largest Subspace Angle with Number of Input Dimensions for Engineering Functions

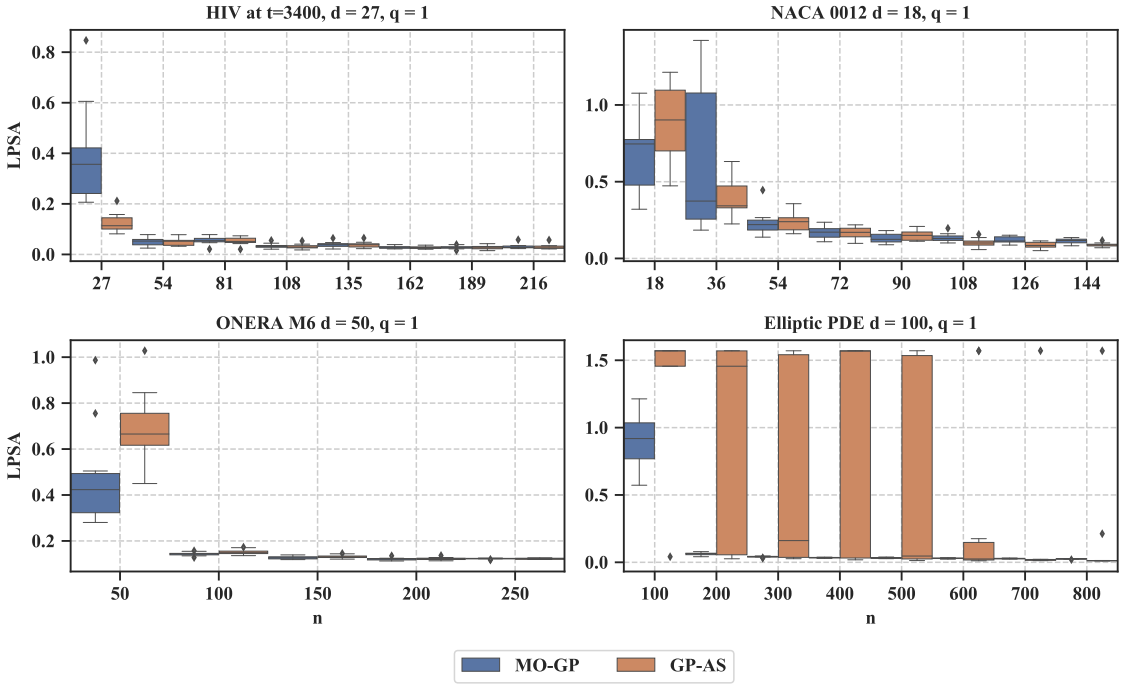


Figure 88: Variation of Largest Subspace Angle with Number of Input Dimensions for Engineering Functions

Figure 88 shows that for the more realistic engineering datasets, similar behavior is observed. For the relatively lower-dimensional HIV and NACA datasets, the recovery of the true AS by both the GP-AS and MO-GP methods is competitive in terms of the LPSA. Although not distinguishable clearly, at larger values of training sample size n, the MO-GP marginally outperforms the GP-AS method for the HIV dataset but is outperformed by GP-AS for the NACA dataset. In the case of the 50- and 100-dimensional ONERA M6 and Elliptic PDE datasets, the GP-AS method is clearly outperformed by the MO-GP method with regard to recovery of the AS for small values of n in-line with the observations so far. Particularly for the Elliptic PDE, note that the GP-AS method struggles to find the AS even at large values of n as indicated by the large interquartile range for the LPSA.

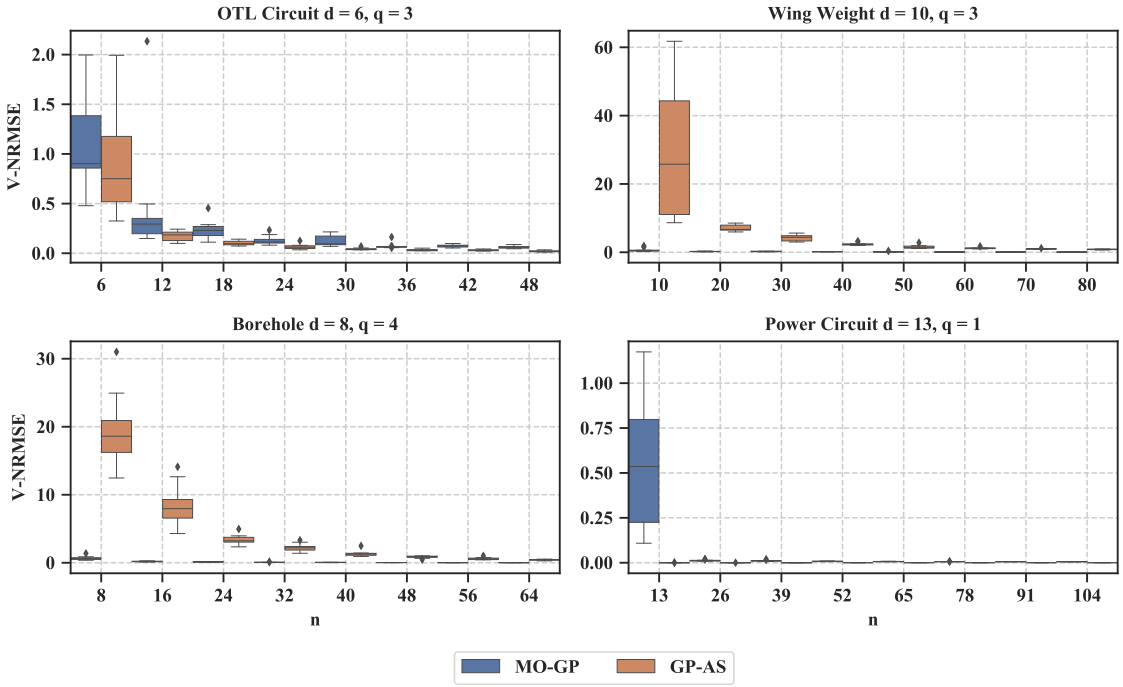


Figure 89: Variation of Validation Normalized Root Mean Squared Error with Number of Training Samples for Engineering Functions

For a more complete picture, let us look at the behavior of the NRMSE shown in figure 89 and figure 90. Except for the NACA 0012 problem, despite having smaller values for the LPSA at small values of n, the MO-GP has a higher prediction error than the GP-AS method. On further inspection of this suspicious behavior, we found that the MO-GP

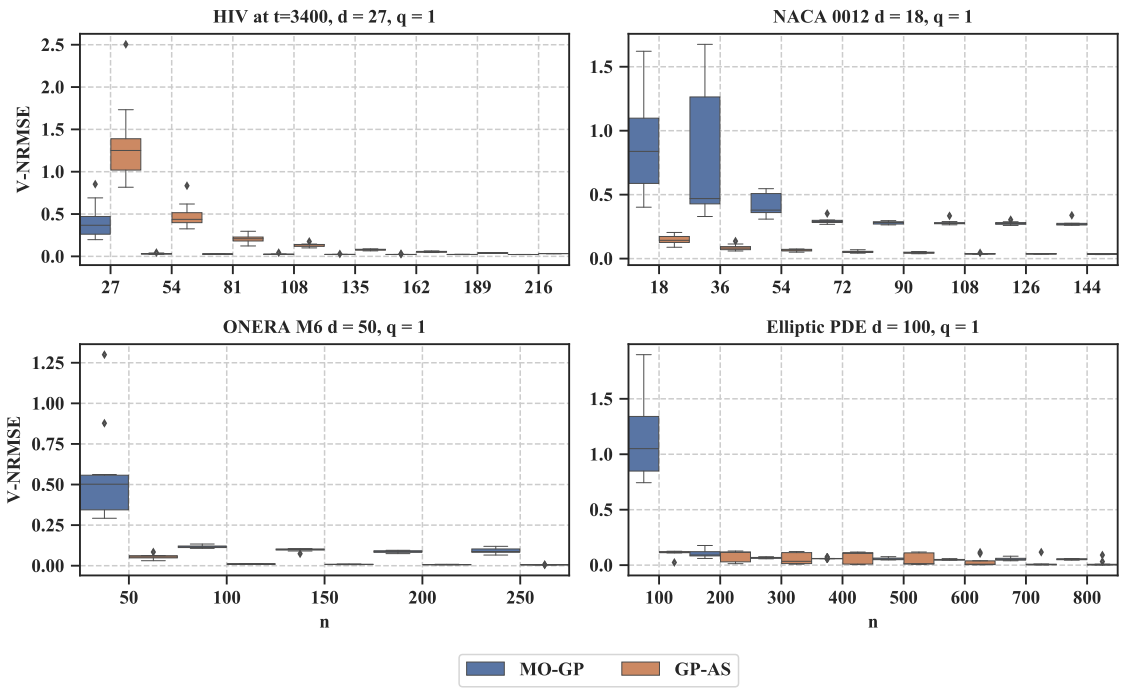


Figure 90: Variation of Validation Normalized Root Mean Squared Error with Number of Training Samples for Engineering Functions

method was overfitting to the training data in the 1-dimensional subspace that it identified. For the GP-AS method however, the training procedure in high-dimensions is known to be ineffective. It so happens that at extremely small values of n, the validation NRMSE for the GP-AS is comparable to its poor training NRMSE. As soon as the number of training samples increases, as expected the MO-GP is either more accurate or more consistent in its ability to predict for all engineering functions and datasets with the exception of the NACA dataset. Even though the NACA dataset has a distinct 1-dimensional AS, the projection of its outputs on the AS is extremely noisy. As a result, even though the MO-GP finds the correct AS, it struggles to fit all the introduced noise using a GP in the projected space. The GP-AS method on the other hand, uses the GP (trained in the original space) to make the predictions after all and therefore does a clearly better job at the prediction task. We will later confirm that when MO-GP is given one additional dimension for the input subspace, it performs significantly better both at finding the AS and predicting for new unseen points. One final remark must be made regarding the engineering datasets.

The HIV, NACA, ONERA M6, and Elliptic PDE datasets all have $q = 1$ and are therefore relatively easier to handle. When $q \geq 1$, the benefits of MO-GP are highlighted even for functions with a modest number of inputs (see fig.87 and fig. 89).

Summary. In this experiment, we performed a thorough study to assess whether the proposed method recovers the true AS and benchmarked its performance against the recently proposed GP-AS method with respect to the input size, AS dimension's size, and number of training samples. It was surprisingly found that the MO-GP method indeed recovers the true AS given input-output pairs without requiring gradients. In general, the MO-GP mostly outperformed the GP-AS method except in a few cases where the performance was competitive. The number of training samples required to successfully recover the AS and obtain a model with good predictive accuracy scales with both the number of inputs d and the size of input subspace q. The value of q was found to have a relatively more significant impact on n than d. The lack of performance in the NACA example highlights a noteworthy disadvantage of simultaneously performing supervised dimension reduction and training the surrogate model. Despite a strong low-dimensional structure, the introduction of a significant amount of noise in the projected space can cause the MO-GP method to struggle.

5.4.3 Setting the Input Subspace Size

Our investigations so far have either varied the dimensionality of the input subspace or have relied on the ASM for setting its value. This research question evaluates whether a commonly used model selection criterion can be used to set the size of the input subspace to be computed by the MO-GP surrogate model. Because the ASM provides a heuristic to set the size of the low-dimensional subspace, it puts onus on any competing method to provide a method to find and set the size of the input projection subspace. The following question addresses the issue of choosing the size of the low-dimensional input projection subspace for the MO-GP:

Research Question 2.3

Given a function to be approximated using the proposed model, how to assess whether a low dimensional input subspace exists? If it does, how can the dimensionality of the low dimensional input subspace be computed?

Whether an AS exists or not is determined by inspecting the spectrum of matrix $\boldsymbol{C}$ for the ASM. While the ASM requires gradients, the proposed method is specifically designed to work with input-output pairs. Unfortunately, there exists no silver bullet in regard to checking the existence and size of a low-dimensional input subspace for a given function using a set of its input-output pairs. As such we must assume that it exists. Fortunately though, most engineering functions in high-dimensions have a low-dimensional structure owing to the Pareto principle which states that 80% of the variation in an output is affected by 20% of all the factors that can affect it.

Conceding to the rather discouraging need to assume that a low-dimensional subspace exists, let us shift our focus to finding its size, given it exists. Recall that unlike the ASM, the gradients are not available to estimate the size of the input projection subspace. Notice however that the input subspace essentially belongs to the set of parameters the MO-GP estimates via the MLE training procedure. Different values of the input subspace dimension q lead to different MO-GP models. Therefore, the problem of estimating the value of q is somewhat equivalent to choosing a suitable model from a set of models trained with a range of q values. This is known as the model selection problem in machine learning parlance. We assert that setting the value of q can be approached via the vantage point of model selection in machine learning as follows:

Hypothesis 2.3: *If a lower dimensional subspace exists for the function being approximated, then choosing its exact size for constructing a surrogate model is a matter of trading predictive accuracy for size of the model and therefore, analyzing a combination of an appropriate model selection criterion and predictive accuracy metric will yield the size of the lower dimensional input subspace such that an acceptable balance between model parsimony*

and predictive accuracy is achieved.

Any learning method's final success depends on its ability to generalize. In other words, a model building exercise is successful if it predicts accurately for data excluded from the training process. As one can imagine, models can be made as complex as possible to fit the data and accurately predict at unseen input values. Striking a desired balance between model parsimony and accuracy is the goal of model selection.

Often, we do not have abundance of data in real-life scenarios to assess the predictive accuracy of models. One popular solution to this problem is to use cross-validation (CV) where the entire dataset is split into k-folds. Then, a model is trained using $k-1$ folds while the k^{th} fold is kept out. Using the k resulting models, an estimate of the model's generalizability can be obtained. However, depending on the size of the folds we may easily be in a situation where a large number of models must be trained.

From a Bayesian standpoint, the problem of model selection can be elegantly explained as follows: Given a dataset $\mathcal{D}$, a family or set of models $\{\mathcal{M}_i\}$ with parameters $\{\boldsymbol{\theta}_i\}$, the likelihood of observing the data given the i^{th} model, and a prior for the corresponding parameters, the model evidence is the likelihood of the data given a model. However, the model evidence is cumbersome to compute by definition. It is given for the i^{th} model as

$$p(\mathcal{D} \mid \mathcal{M}_i) = \int p(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M}_i) p(\boldsymbol{\theta} \mid \mathcal{M}_i) \mathrm{d}\boldsymbol{\theta} \tag{109}$$

The Bayesian approach to model selection involves choosing the model with the highest probability given the observed/training data. The posterior probability of a model can be written as follows using the Bayes formula

$$\begin{aligned} p\left(\mathcal{M}_m \mid \mathcal{D}\right) &\propto p\left(\mathcal{M}_i\right) \cdot p\left(\mathcal{D} \mid \mathcal{M}_i\right) \\ &\propto p\left(\mathcal{M}_i\right) \cdot \int p\left(\mathcal{D} \mid \boldsymbol{\theta}, \mathcal{M}_i\right) p\left(\boldsymbol{\theta} \mid \mathcal{M}_i\right) d\boldsymbol{\theta} \end{aligned} \tag{110}$$

Since the evaluation of the integral (or the evidence) is intractable, we often resort to approximations of the posterior probability of a model conditioned on the training data, also called model selection criteria. Information criteria also effectively penalize the models for having an excessive number of parameters. The Bayesian Information Criterion [99]

(BIC) and the Akaike Information Criterion [4] (AIC) are two of the most successful and commonly used criteria for model selection under this paradigm.

The choice between AIC and BIC for model selection has no particular winner. We adopt the BIC because unlike AIC, it is asymptotically consistent as a selection criterion. Stated in simpler terms, the BIC chooses the best model among a family of models with a probability of one as the number of training samples $n \to \infty$. While the AIC tends to choose models that are overly complex for large n, the BIC typically chooses simpler models [115] (sometimes too simple). In our case, the BIC can be used a model selection criterion, treating the set of models obtained for a range of sizes q of the low-dimensional subspace as a family. With a large set of training samples, a model for each setting of q can be trained to compute the BIC given as

$$\text{BIC}(q) = \log \mathcal{L} - \frac{1}{2}(\log n) \cdot n_{\boldsymbol{\theta}} \tag{111}$$

where $\mathcal{L}$ is the log-marginal likelihood of the trained MO-GP model with a q-dimensional projection subspace, n is the number of training samples, and $n_{\boldsymbol{\theta}}$ is the number of parameters of the MO-GP model written as

$$n_{\boldsymbol{\theta}} = 2 + q + d \times (d - q) \tag{112}$$

where the first two terms correspond to the hyperparameters of the GP and the second term corresponds to the number of parameters for a q-dimensional Grassmann manifold.

Note that we must proceed with caution with the BIC because it is an approximation and its accuracy depends on the number of training samples relative to the number of parameters. While we desire parsimonious models which the BIC naturally prefers, a situation where too much accuracy is traded for simplicity must be avoided. Therefore, we propose the use of a combination of the BIC and the validation NRMSE to choose the value of q. The V-NRMSE metric serves the purpose of 1) serving as yet another indicator of the gain in the predictive accuracy and 2) giving a faithful estimate of model performance in cases where the BIC cannot be trusted. Our choice of expression for the BIC yields the best model when the BIC is maximized. Also note that even though the model selection approach involves training a model for each value of q, it is much more efficient that the CV approach

which requires multiple training repetitions over partitions of the dataset. Therefore, it is our preferred choice. In order to observe the behavior we expect to empirically verify our hypothesis, both metrics for this experiment are plotted against the value of q.

Details of the Experiment. The plan for this experiment follows the steps for the experiment 2.1 (see fig. 77). On a subset of the scalar functions listed in Table 5 with a known AS, models with a range of values of q are trained with a sample size ($n = 450$) that is sufficient for training a good model (i.e. one that recovers the true AS and does not over-fit). We perform 10-20 repetitions with randomly sampled training sets for each setting of q and record both the BIC and the validation NRMSE.

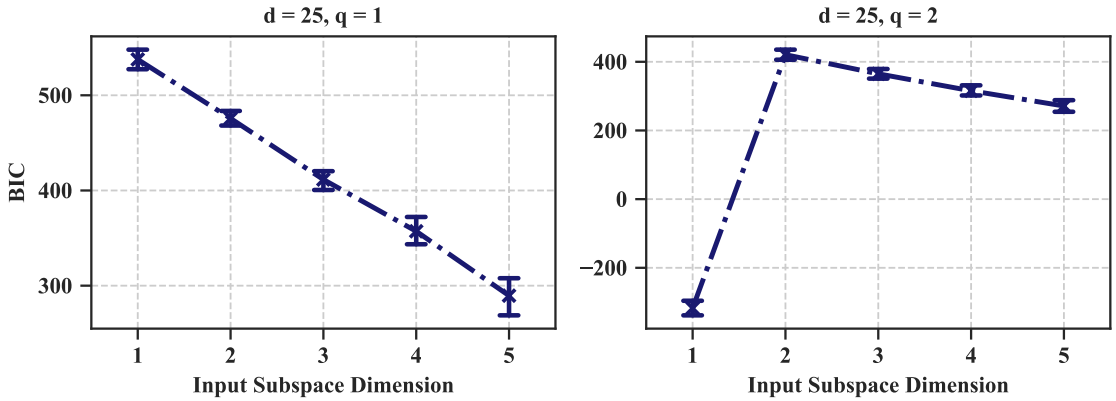


Figure 91: Variation of the Bayesian Information Criterion with Number of Input Dimensions for the 25 Dimensional Quadratic Problem

Results for the Quadratic Functions with $d = 25$. Figure 91 shows the progression of BIC as the value of q is increased. It clearly shows the effectiveness of the BIC when the function is actually low-dimensional (by construction, in our case). For the 25-dimensional quadratic function with 1- and 2-dimensional AS, the BIC distinctly takes its maximum value exactly at the true input space dimensionality beyond which the BIC decreases. We refer to equation (111) to explain what this behavior. For q values larger than the size of the true AS, the second term, i.e. the penalty for complex models, starts increasing faster than the first term, i.e. the likelihood of observing the data. On inspection of the model performance using the V-NRMSE in figure 92, we see that the best performing

models correspond to $q = 1$ and $q = 2$. Recall that the quadratic functions have artificially introduced noise to emulate the consequence of projecting the inputs on the outputs for real-life functions. Inclusion of dimensions beyond the true value of q causes the model to fit the noise in the training data. As a result, the T-NRMSE decreases (see Fig. 78) but the generalizability increases; this is a classic example of overfitting.

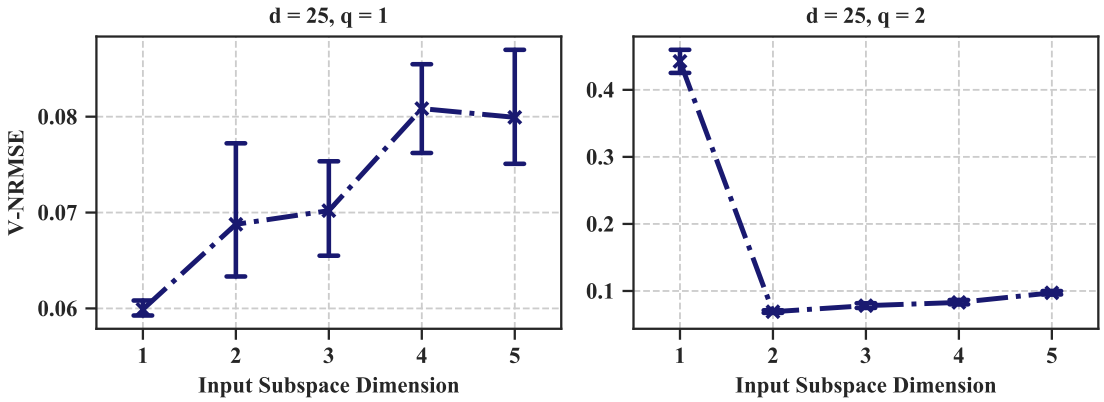


Figure 92: Variation of Validation Normalized Root Mean Squared Error with Number of Input Dimensions for the 25 Dimensional Quadratic Problem

Results for Engineering Functions and Datasets. The rather clear recommendation for the value of q in the case of the quadratics becomes somewhat murky when considering real datasets and functions that do not have an exactly low-dimensional structure. In fact, we argue that the notion of the true value of the AS does not exist with engineering functions. As we saw before, the ASM method also relies on a heuristic that essentially places a bound on the acceptable loss in accuracy by inspecting the spectrum of $\boldsymbol{C}$.

This is another reason why we recommend the use of V-NRMSE in addition to the BIC in deciding a good value for q as shown in figure 93 and figure 94. For the analytical wing weight function, note that the BIC does not show a sudden drop beyond its AS dimension $q = 3$. The reason is clear in Figure 94 where it can be seen that the V-NRMSE decreases by approximately one-hundredth of the standard deviation (noise in the validation set) between $q = 3$ and $q = 4$. The V-NRMSE shows that there is further scope for improvement. However, for the borehole function there is a clear drop in the BIC beyond $q = 4$ implying

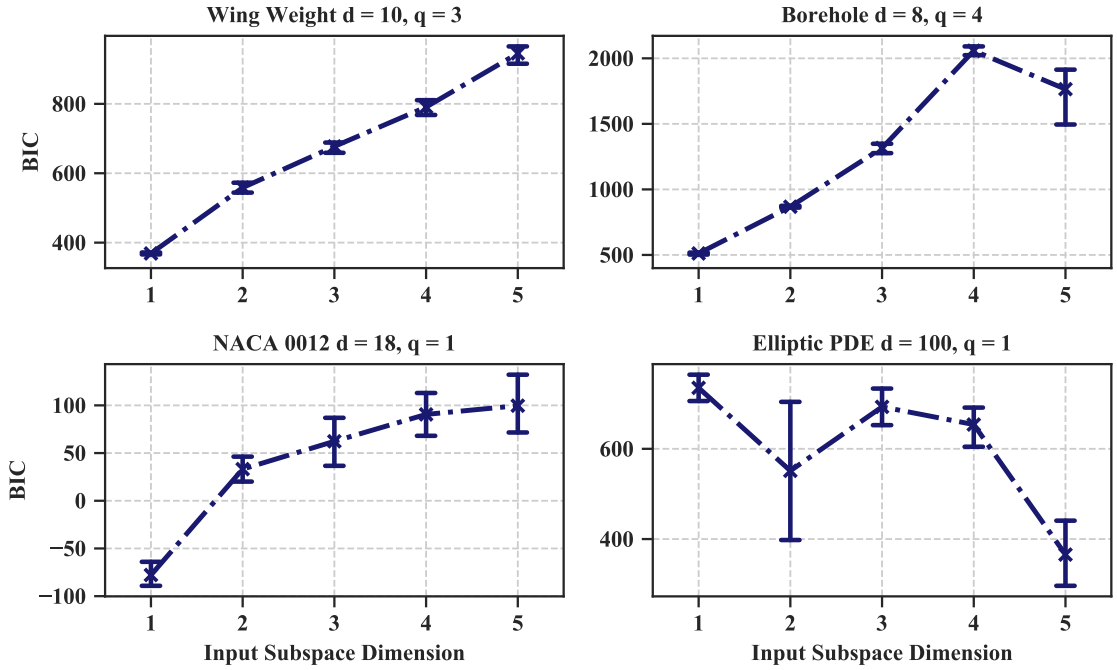


Figure 93: Variation of the Bayesian Information Criterion with Number of Input Dimensions for Engineering Problems

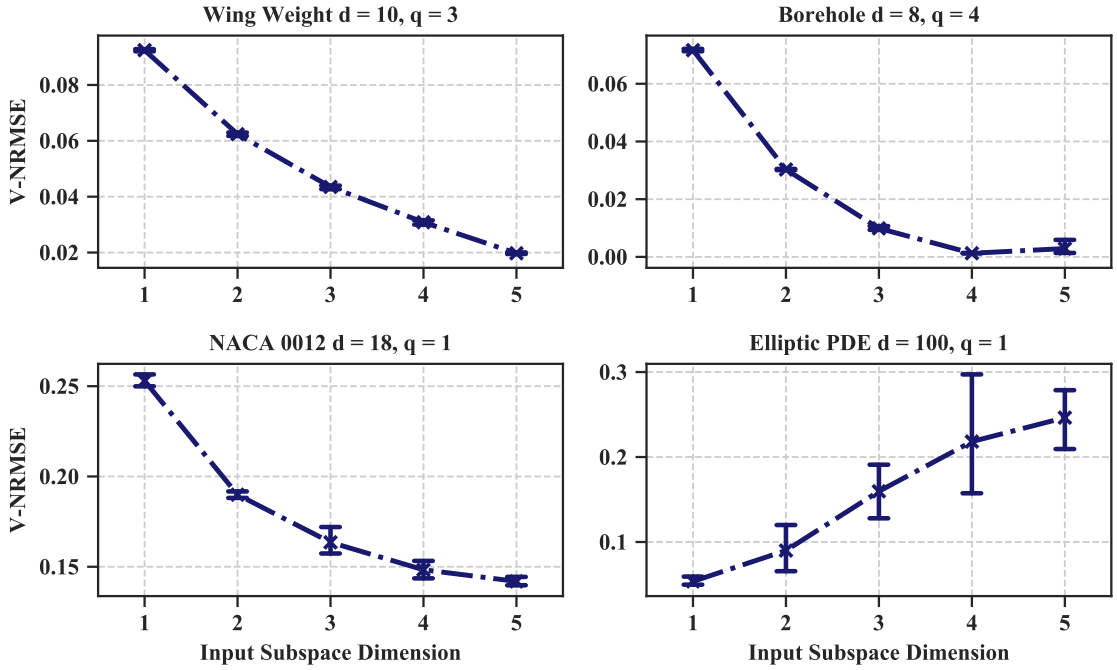


Figure 94: Variation of Validation Normalized Root Mean Squared Error with Number of Input Dimensions for Engineering Problems

that MO-GP has started to capture the noise introduced due to the projection onto a lower dimensional subspace.

Recall that in the previous experiment, the MO-GP model did not perform well on the NACA 0012 dataset with $q = 1$. The reason becomes clear when one sees the behavior of the BIC and the V-NRMSE with increasing q. Note the sudden rise and accompanying drop in the values of BIC and V-NRMSE respectively, between $q = 1$ and $q = 2$. The V-NRMSE drops by five-hundredths of the standard deviation indicating the loss in accuracy incurred for having $q = 1$ instead of $q = 2$. If the gains in accuracy are deemed minor for the specific application, then $q = 1$ is indeed sufficient. For the elliptic PDE problem, the behavior of BIC and V-NRMSE are similar to that of the quadratic functions which indicates that the problem is indeed 1-dimensional. In fact, the model starts overfitting the noise beyond $q = 1$.

Summary. The results for this experiment brought some important observations regarding the choice of the low-dimensional subspace q. The BIC alone is a good indicator for selecting the correct value of q only for exactly low-dimensional functions. However, in the general case of low-dimensional functions in high-dimensions, the choice of q reduces to the trade-off between model parsimony and model generalizability. Of course, the statement is valid only if the function has a low-dimensional structure which is unfortunately an assumption. We recommend the use of both the BIC and the V-NRMSE because it was found that while BIC approximately balances parsimony and likelihood of observing the data, the V-NRMSE actually indicates the predictive accuracy for unseen points.

5.4.4 POD and Interpolation-Based ROMs with MO-GP

Now that we have a decent understanding of the MO-GP model, we are ready to perform studies, similar in vein to the ones done for scalars, on its application to construct POD and interpolation-based ROMs (as described in section 5.2). We hope to achieve it through the following question:

Research Question 2.4

For constructing POD and interpolation-based ROMs, does the application of the developed model to identify a low-dimensional subspace and train surrogates for the map between the POD coefficients and the inputs yield models with good predictive performance at the field output level?

In section 5.2, we proposed the use of MO-GP to fit the map between the POD coefficients and the high-dimensional inputs. The previous experiments in this research area have demonstrated the effectiveness of the MO-GP method in discovering an input subspace similar to the AS using input-output pairs exclusively. Recall that following the dimension reduction step, the POD and interpolation method trains data-fit surrogate models to encode parametric dependence of the low-dimensional representation (POD subspace coordinates) of the field outputs. Therefore, it stands to reason that such data-fit surrogate models for the POD subspace coordinates can be constructed using the MO-GP regression when the inputs are high-dimensional. Whether or not the variation in the POD subspace can be captured using low-dimensional input subspaces will finally determine whether the MO-GP is successful in creating a surrogate. However, because the POD modes (relatively few in number for low-rank systems) capture statistically important features of the field quantity, it is reasonable to expect a few important directions in the input space to explain the variation of at least the POD coefficients corresponding to the first few modes. Fortunately, the BIC and V-NRMSE can be relied on for verifying whether as AS exists and if it does, find an appropriate size. In hypothesis form, it may be stated as follows:

Hypothesis 2.4: *If there exist low-rank input subspaces that capture a majority of the variation of the respective coordinates in the low-rank POD subspace, then the proposed method can be used to simultaneously identify the low-dimensional input subspace and train surrogates for each of the POD coordinates.*

Although the use of MO-GP was found to be successful for scalar functions in the previous experiments, their ability to find the true AS, if one exists, must be verified when

used with POD and interpolation-based ROMs involving high-dimensional inputs. Several reasons warrant performing a thorough study similar to the one done for scalar functions: 1) In contrast to scalar functions defined in high-dimensional input spaces, the POD and interpolation method assumes that the POD subspace coordinates are smooth functions of the inputs. 2) the POD modes corresponding to the small singular values capture the noise in the outputs by construction. Whether a low-dimensional input subspace exists for them and whether it can be found by the MO-GP model is unknown. 3) In addition to the aforementioned points, the effectiveness of the MO-GP as the problems change in output and input size must also be characterized to gain a thorough insight and provide recommendations for its use with POD and interpolation-based ROMs.

The experiment for this research question first performs a complete characterization of the use of MO-GP with POD and interpolation-based ROMs on the task of constructing an emulator for the field output from the high-dimensional elliptic PDE problem detailed in section 5.3.2.1. Recall that we deliberately created three variants of this problem with 25, 50, and 100 input parameters to test the sensitivity of the proposed methodology. In addition, the true AS for the POD coefficients were computed using approximate gradients obtained by projecting the Jacobians of the field output (obtained intrusively) onto the truncated POD basis. Doing so allows for a thorough comparison of the subspace computed by MO-GP with the AS of each of the POD coefficients. Once the use of MO-GP with POD and interpolation for problems with high-dimensional inputs and outputs is whetted, we demonstrate its use for the construction of ROMs for two practical airfoil problem (see sec. 5.3.2.2) and one CRM wing problem (see sec. 5.3.2.3).

For each POD coefficient, we use the discrepancy between the computed subspace and the true AS as indicated by the LPSA metric (eq. (52)) to study the effectiveness of the MO-GP in recovery of the low-dimensional projection subspace. The predictive accuracy of the GP surrogates trained in the identified input subspace for each POD coefficient is measured using the NRMSE (eq. (106)). Since the final goal of any ROM is the prediction of a high-dimensional output vector representing a spatially (and/or temporally) distributed quantity, the success of the proposed methodology must be evaluated at the field output level.

The two main sources of error are the projection onto the POD basis and the subsequent regression in the POD subspace. It is instructive to decompose the total prediction error at the field level into contributions from projection and regression respectively. The total error at the field level for a sample solution is defined as

$$\epsilon_{\text{total}} = \frac{\|\boldsymbol{w}_{\text{actual}} - \boldsymbol{w}_{\text{predicted}}\|_2}{\|\boldsymbol{w}_{\text{actual}}\|_2} \tag{113}$$

The error due to projection onto the POD basis is written as

$$\epsilon_{\text{projection}} = \frac{\|(\boldsymbol{I} - \boldsymbol{\Phi}_k \boldsymbol{\Phi}_k^T)\boldsymbol{w}_{\text{actual}}\|_2}{\|\boldsymbol{w}_{\text{actual}}\|_2} \tag{114}$$

The regression error at the field level (i.e. obtained by expanding in the POD basis) can be written as

$$\epsilon_{\text{regression}} = \frac{\|\boldsymbol{\Phi}_k(\boldsymbol{\Phi}_k^T \boldsymbol{w}_{\text{actual}} - \boldsymbol{h}(\boldsymbol{\mu}_{\text{test}})\|_2}{\|\boldsymbol{w}_{\text{actual}}\|_2} \tag{115}$$

where $\boldsymbol{h} : \mathbb{R}^d \to \mathbb{R}^k$ is the vector-valued function that maps the parameters to the POD coordinates, each component of which is trained using the MO-GP model. The total can also be expressed in terms of a component perpendicular (eq. (114)) to the POD subspace and a component contained (eq. (115)) in the POD subspace as

$$\epsilon_{\text{total}} = \sqrt{\epsilon_{\text{projection}}^2 + \epsilon_{\text{regression}}^2} \tag{116}$$

The decomposition of the error at the field level provides insight into the performance of the field prediction task by delineating various sources of error. For instance, if the projection error is consistently larger than the regression error, it may be sensible to increase the number of retained POD modes and if the regression error is larger on average, it indicates that the surrogates in the latent space must be improved.

Details of the Experiment. This experiment can naturally be divided into two parts. The first part involves finding the value of q for each POD coefficient. The second part involves training the POD and interpolation model using q_i-dimensional MO-GP models for the i^{th} coefficient.

The first part begins with the generation and collection of snapshots for an appropriate space-filling DoE such as LHS (used here) by evaluating the expensive high-fidelity analysis.

Then, the deterministic SVD is performed to obtain the truncated rank-k POD basis for some prescribed RIC (99.9% in our case). Following projection of the snapshot matrix onto the POD basis, the dataset for the projected coordinates is used to train MO-GP models with varying sizes of the low-dimensional input subspace q for each POD coordinate. Note that we do not repeat the comparisons with the benchmark GP-AS method because the previous experiments have clearly shown the benefits of using the MO-GP instead. The BIC and V-NRMSE are computed and recorded for each MO-GP model. Figure 95 shows a summary of the steps for this part. We report results over 5 repetitions with randomly sampled training sets. These values are used to first check whether a low-dimensional input

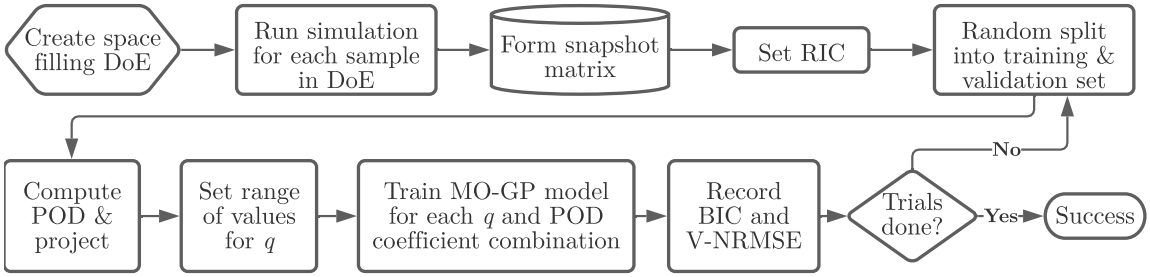


Figure 95: Setup for Experiment 2.4, Part A

subspace exists for each coefficient and if it does, set its size q_i. Note that for the elliptic PDE problem, the sizes of the actual AS are available for each POD coefficient. Our BIC and V-NRMSE based technique's accuracy should ideally provide similar estimates.

The second part of this experiment proceeds with the construction of ROMs with appropriately sized input subspaces (determined from previous experiment) for each coefficient. In the case of the elliptic PDE test case, we measure the LPSA between the actual and the computed subspace in addition to the averaged total error (definition for a sample in (eq. (113))) and regression error (definition for a sample in (eq. (115))) at the field level for both the training and validation sets. Note that the projection error is not reported because it is irreducible given an RIC and training dataset size. From now on, the mean validation (training) total error and regression error are denoted as V- (T-) MTE and V- (T-) MRE respectively. We report results over 10 repetitions with randomly sampled training sets.

An important remark regarding the ability to test the sensitivity of the effectiveness of MO-GP with respect to the number of training samples must be noted before proceeding. The size of the training set when constructing ROMs is typically set such that the projection error for a given RIC is smaller than some acceptable tolerance over the entire dataset (training and validation sets) or it stops changing significantly when additional training points are added. It is then assumed that the training dataset thus obtained will yield basis vectors that span the entire space of solutions on compression. As such, this places a minimum requirement on the sizes of training datasets that can be used for obtaining ROMs with small projection errors. Consequently, the resulting range (and number) of training sample sizes on which studies are performed is smaller for construction of ROMs in contrast to the case of scalars.

Results for the Elliptic PDE Problem. The reader is encouraged to recall and get familiarized with the details of the three variants of the elliptic PDE problem (see sec. 5.3.2.1) designed to characterize the functioning of the proposed methodology. Figure 70 shows that attaining an RIC of 99.9% for all the elliptic PDE problems requires a rank-6 truncation of the POD basis.

As shown in figure 71, the approximations of $\boldsymbol{C}$ obtained for each the 25-, 50- and 100-dimensional problem's POD coordinates' gradients display a sharp drop in their eigenvalue spectrums indicating the presence of a low-dimensional input structure. To obtain the actual value of q for each POD coordinate's AS, we choose the index in the spectrum of the Monte-Carlo approximation of $\boldsymbol{C}$ that corresponds to an order of magnitude decrease in the eigenvalues. Table 6 shows the resulting sizes of the AS for each of the POD coefficients. Note that we see a clear reduction in the size of the input space for each POD coefficient. Even the later POD modes that capture the nuances in the field vary strongly in an input space that is at most (for the problem with $d = 25$) roughly one-sixth the size of the true input space dimension d.

Figure 96, figure 97, and figure 98 show the variation of the BIC for the MO-GP model as the value of q is increased for 5 randomly sampled training datasets containing 600

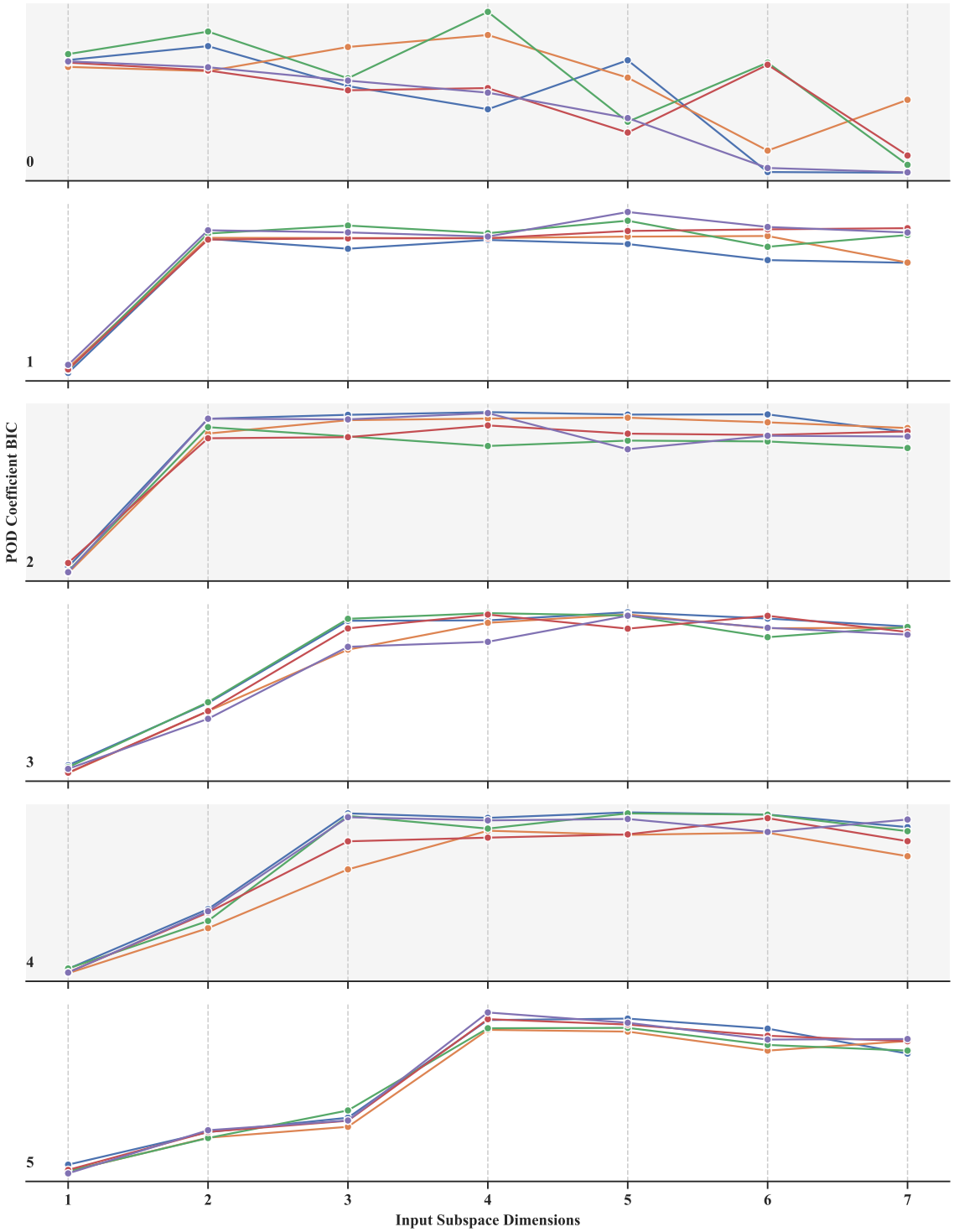


Figure 96: Variation of BIC for the Elliptic PDE Problem with $d = 25$. Colors Represent Different Repetitions.

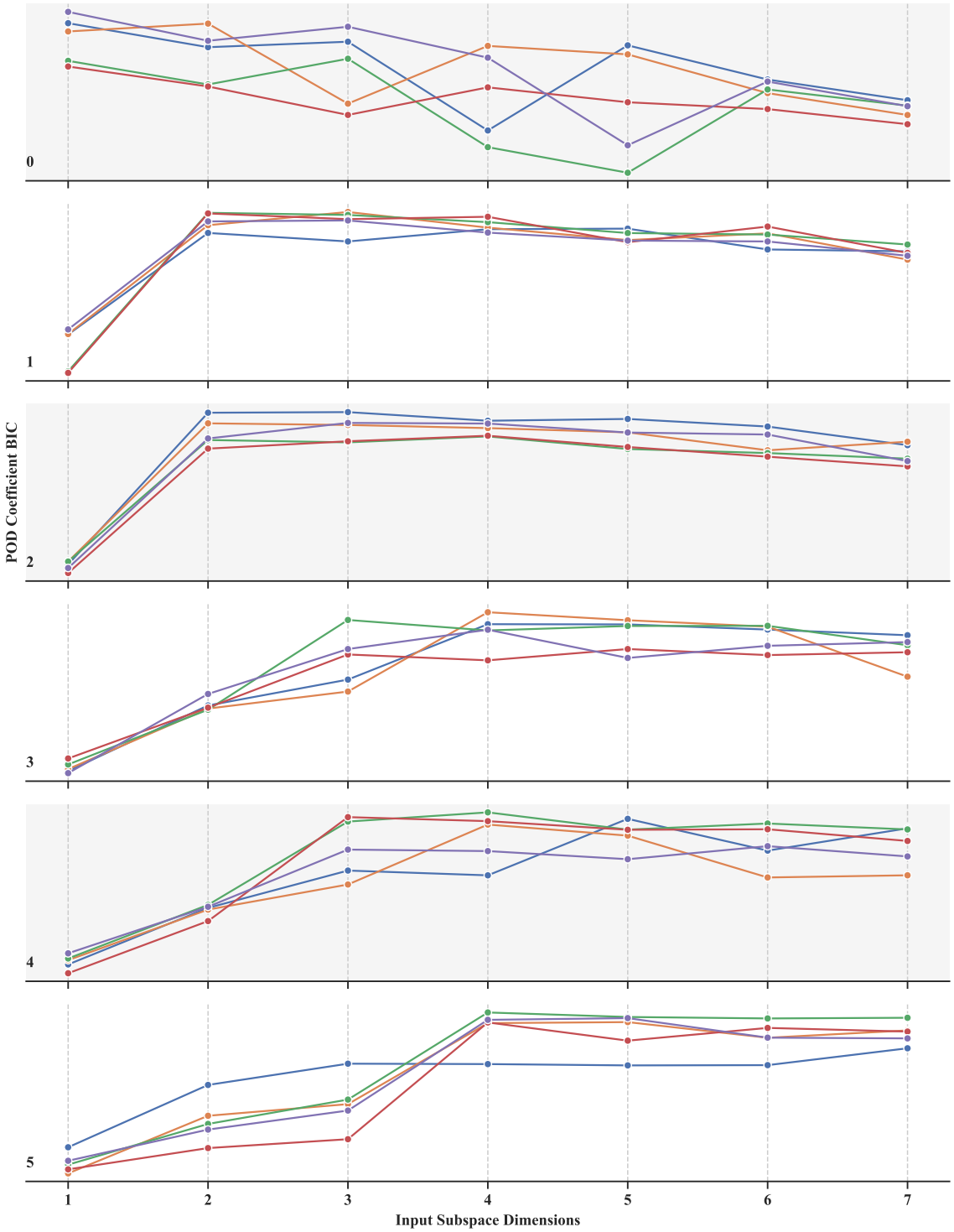


Figure 97: Variation of BIC for the Elliptic PDE Problem with $d = 50$. Colors Represent Different Repetitions.

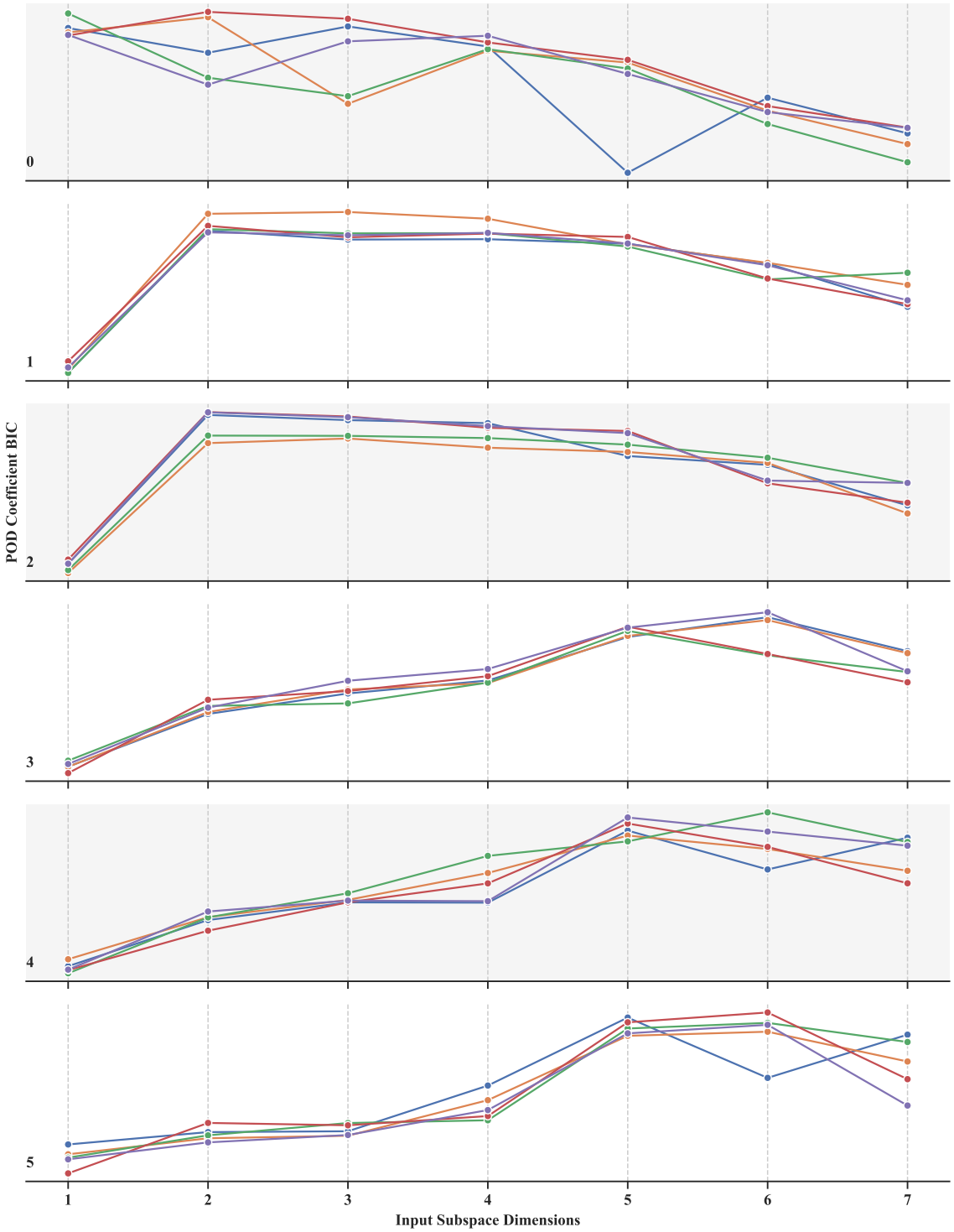


Figure 98: Variation of BIC for the Elliptic PDE Problem with $d = 100$. Colors Represent Different Repetitions.

samples each. Each model is trained with $n_{\text{restarts}} = 500$. While 300 samples are sufficient for a good projection error because of the extremely low-rank structure of the problem, we deliberately choose to present the results obtained for a large training size to avoid the pitfalls of inaccurate computation of the BIC. The figures clearly show that for the problems with $d = 25,\ 50,$ and 100, the BIC is maximized for the first 3 coefficients at values of q equal to their true AS sizes respectively (see Table 6). However, for problems with $d = 50$ and 100, the BIC peaks at values of q larger than the true size of the AS for the last 3 coefficients that capture the relatively noisier and nuanced features in the field. On the other hand, for the relatively smaller problem with $d = 25$, we see that the BIC consistently peaks exactly at the expected value of q, even for the higher POD modes. Several factors give rise to this behavior. We provide reasons based on the observations made for the MO-GP's performance for scalar functions. First and foremost, the optimization problem for the MO-GP becomes progressively challenging to solve numerically as d increases as we saw in the case of quadratic scalar functions. Better performance may probably be a matter of setting a higher number of restarts n_{restarts} and incurring a larger training cost as a result. Second, the already difficult goal of learning the subtle and fine details captured by the later POD modes in a low-dimensional input subspace is exacerbated by the inclusion of a larger number of design parameters. Consequently, the BIC profiles for problems $d \geq 50$ peak at higher values of q implying that the added input dimensions can indeed help in capturing the noise in the higher POD modes. The results are largely in-line with our expectations from the experiments on scalars. The BIC is indeed a good indicator for the selection of the size of the low-dimensional input subspace for the POD coefficients' MO-GP models. We proceed with q values as given in Table 6 for the construction of the ROMs.

The results for the POD and interpolation-based ROMs' predictive performance are reported over 10 trials using randomly sampled training datasets of size 300, 450, and 600 for all the elliptic PDE problems. Because this problem is extremely low-rank in nature, 200-300 samples were sufficient to get small projection errors. In fact, the projection error decreases only marginally beyond 450 training samples even for the problem with $d = 100$.

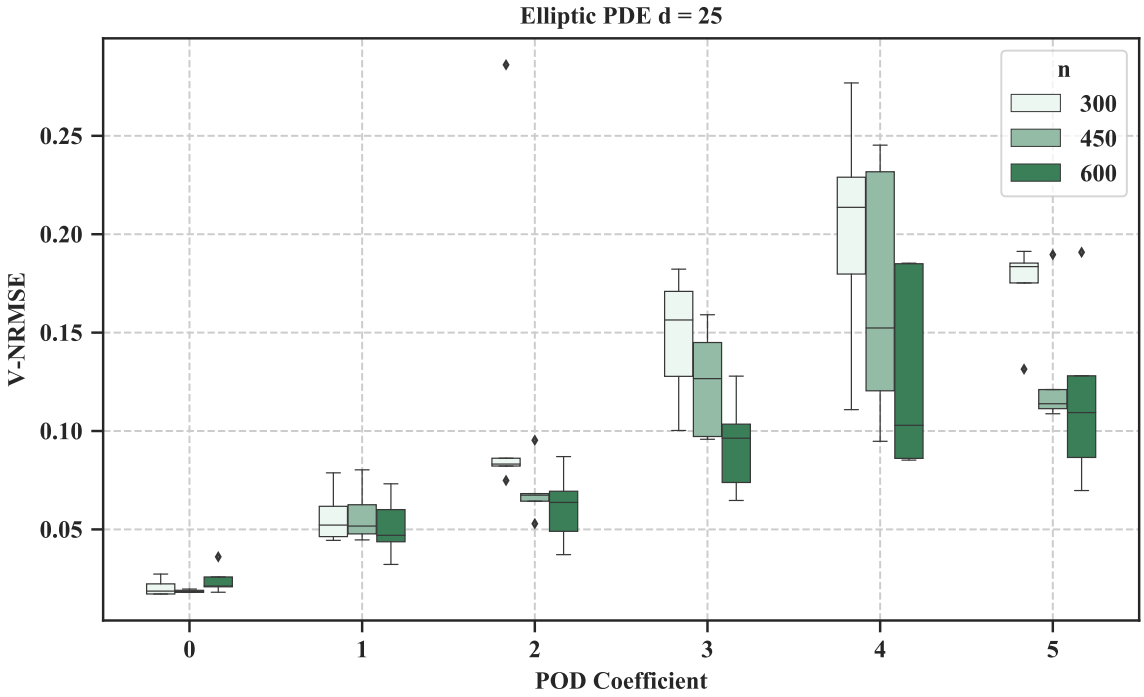


Figure 99: POD Coefficients' Validation NRMSE for the 25 Variable Elliptic PDE Problem

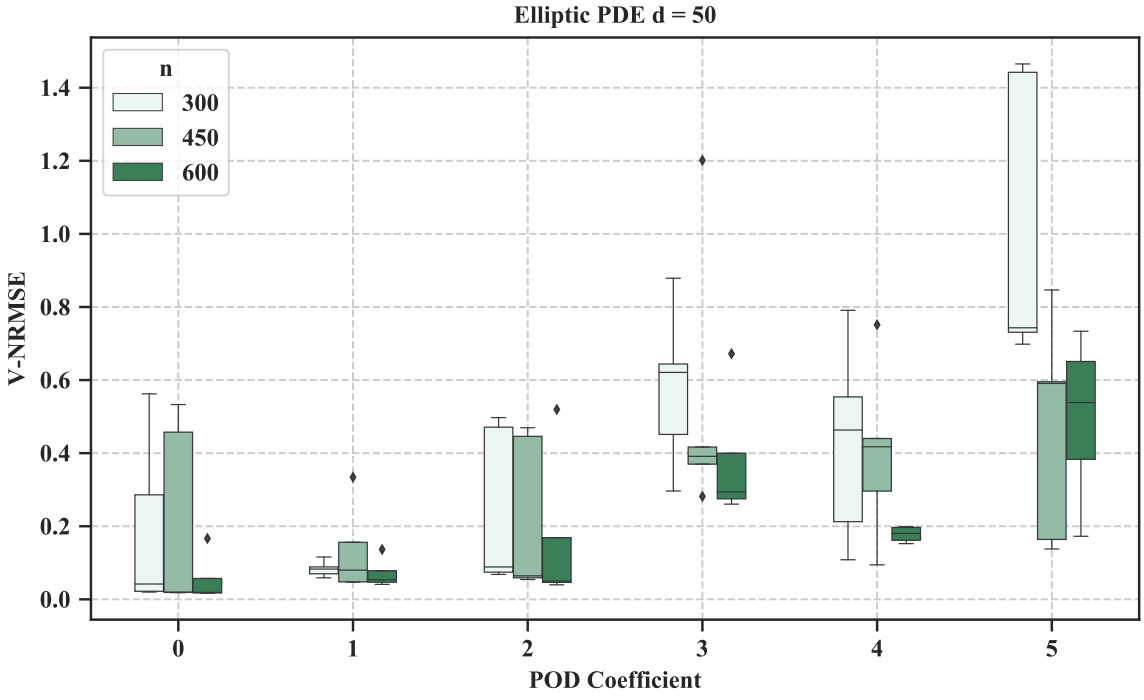


Figure 100: POD Coefficients' Validation NRMSE for the 50 Variable Elliptic PDE Problem

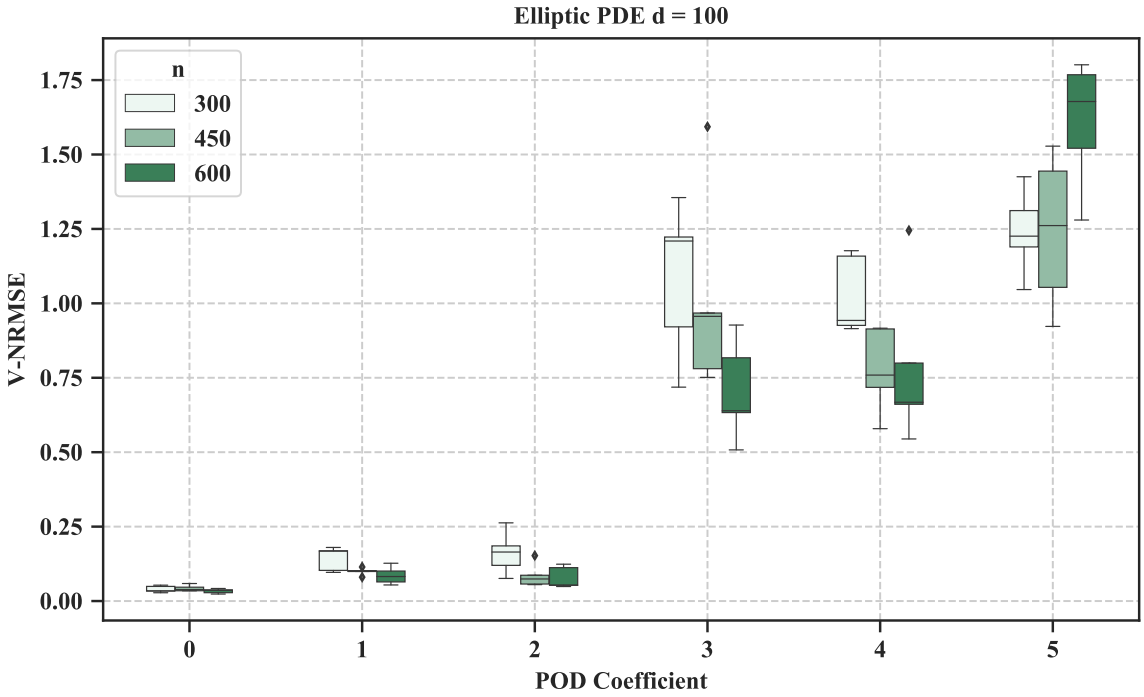


Figure 101: POD Coefficients' Validation NRMSE for the 100 Variable Elliptic PDE Problem

Before presenting the predictions at the field level, let us analyze the predictive accuracy for the coefficients as shown in figure 99, figure 100, and figure 101 for the problems with $d = 25,\ 50,$ and 100 respectively. The first observation concerns the conspicuous decrease in the predictive accuracy across all training samples and all problems as the POD mode coefficient index increases. One obvious reason lies in the fact that the sheer number of parameters in the MO-GP model increases because of the larger AS dimension. Also, as stated before, the later modes start capturing subtle and fine features of the field and hence appear noisy. Additionally, recall that the POD coefficients are centered at zero by definition. Their small noisy values drive the sample standard deviation to values very close to zero. Consequently, even small errors in prediction are penalized heavily when normalized by the almost zero sample standard deviation. Especially when the singular value spectrum of the snapshot matrix decays rapidly, the later modes' contribution to the overall reconstruction is negligible compared to the initial modes. Therefore, the apparently alarming behavior is actually inconsequential at the field level as we will see in detail later.

In general, observe that the predictive accuracy for the coefficients improves as the number of training samples available for training increases. On closer examination, the V-NRMSE values are relatively more consistent for the few initial modes at higher training sample sizes. This behavior is not always observed for the later modes because the surrogate modeling itself becomes more challenging due to presence of noise. Finally, note that in going from $d = 25$ (see fig. 99) to $d = 100$ (see fig. 101), there is a marked decline in the predictive performance of the MO-GP models for the later coefficients. This behavior is consistent with our observation for the values of the BIC. Interestingly, even increasing the value of the corresponding q only increased the accuracy marginally. Assuming that the optimization of the log-marginal likelihood converged to the optimum, the aforementioned observation corroborates the presence of inherent noise in the later POD modes.

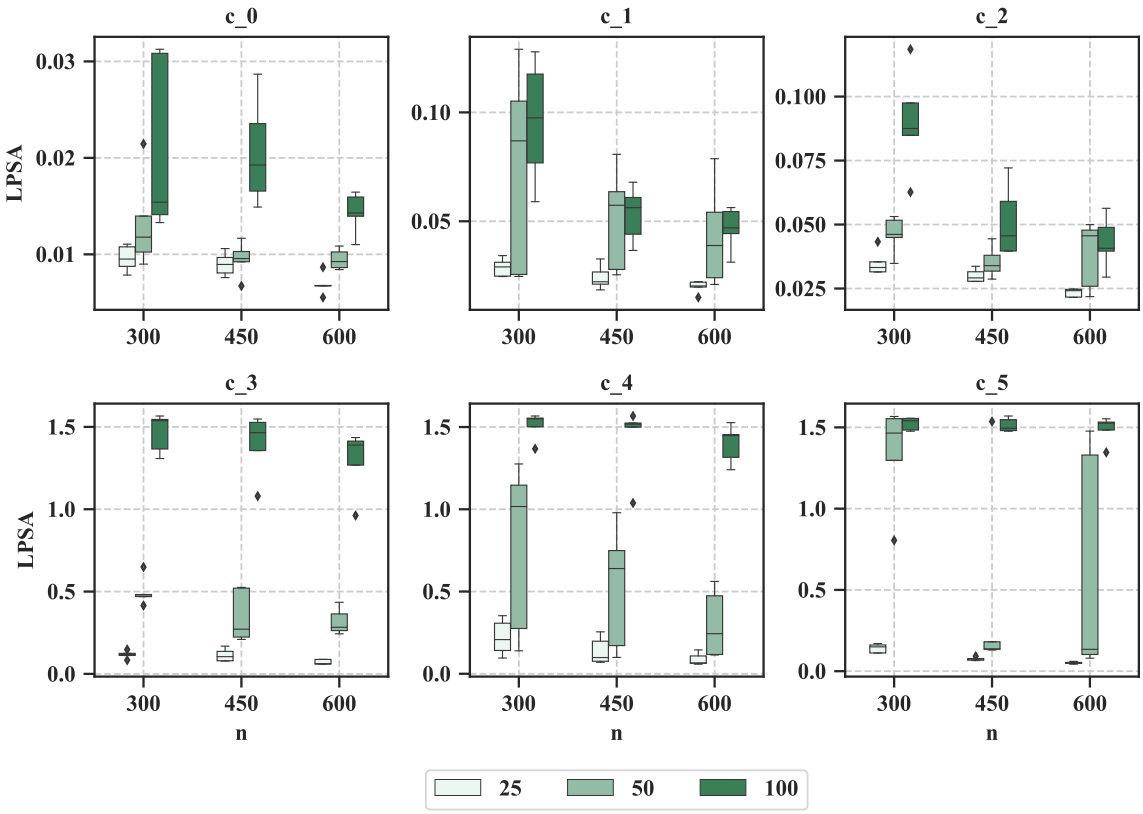


Figure 102: POD Coefficients' Largest Principal Subspace Angle for the Elliptic PDE Problem

The ability of the MO-GP model to find input subspaces for the POD coefficients that are identical to the respective active subspaces as indicated by the LPSA is shown in figure 102. Note that the results largely mirror the ones for scalar functions. In general, it is easier to find the true AS for the POD coefficients corresponding to the dominant modes. However, we must bear in mind that the size of the AS is also smaller for the initial POD modes. Problems with $d \geq 50$ require a larger number of training samples to find the true AS because q increases for the later POD coefficients making the optimization problem more challenging. Additionally, the data also becomes naturally noisy as mentioned before. Despite providing a large number of samples, the MO-GP is unable to exactly find the true AS for the later POD coefficients. Even when $d = 25$, increasing the number of samples improves the consistency with which the true AS is found as indicated by the narrow interquartile ranges across all the POD coefficients. But as pointed out previously for the scalar quadratic datasets, there are instances where the LPSA is large even though the predictions are satisfactory. The explanation in such cases is that the large discrepancy in angles arises for only a few directions in the input subspace while the other, and more important directions may be estimated accurately.

So far we have observed that the MO-GP successfully finds a low-dimensional subspace and gives sufficiently accurate predictions for the POD coefficients. Figure 103 shows the accuracy of the resulting predictive ROMs at the field level. The trends largely resemble the results seen for the coefficient fits. Note the distinct difference between the results for the 100-dimensional problem and the 25- and 50-dimensional problem. On average, the problem with $d = 100$ is 2-6 times more inaccurate when compared to the problems with $d = 25$ and $d = 50$. As expected, the difference in accuracy is larger for small n but starts becoming insignificant for larger n. This occurs due to the relatively poorer coefficient fits for the higher-dimensional elliptic PDE problem especially when the number of training samples is insufficient for the size of the AS. A reassuring trend is the rise in the consistency of the training and validation MRE and MTE as the number of training samples increases. Additionally, the similarity in the values of the MTE and the MRE indicates that the regression error has a larger contribution than the projection error to

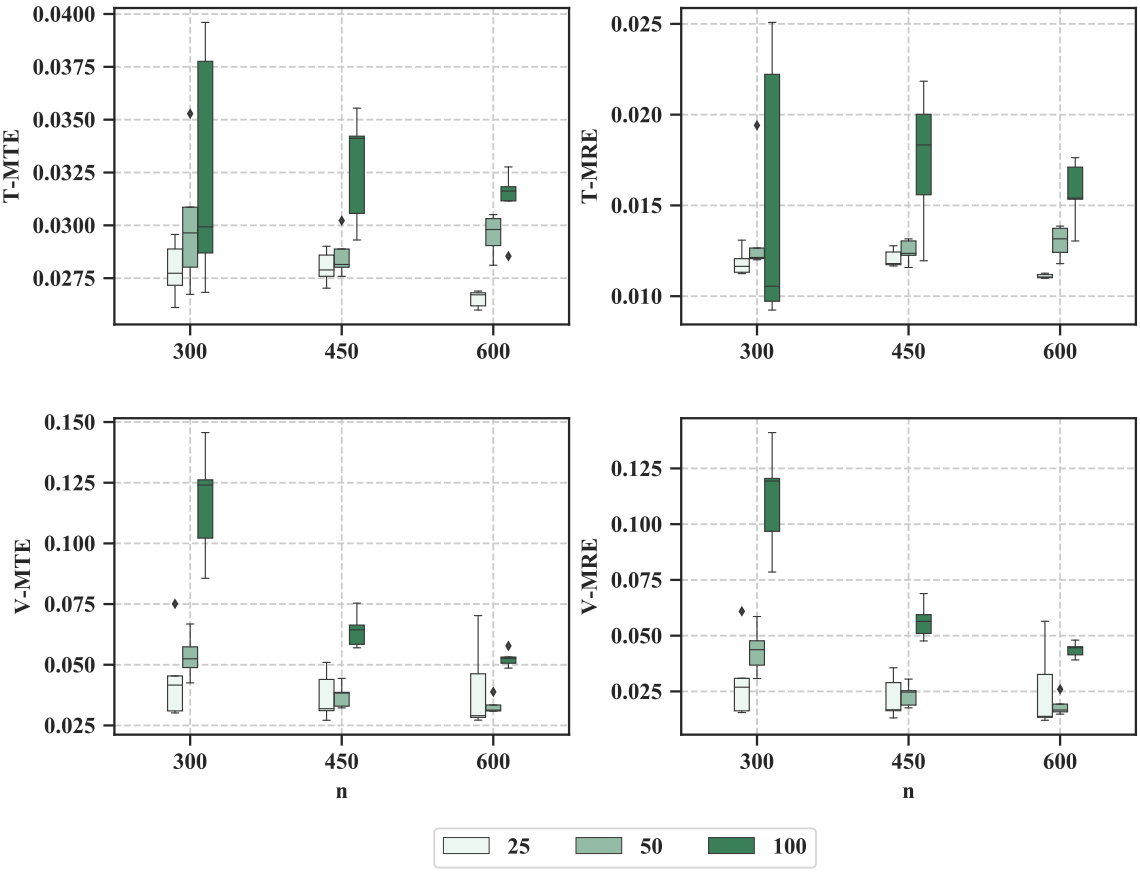


Figure 103: Field Error Metrics for the Elliptic PDE Problem

the total error. The most important takeaway from figure 103 is the observation that even for the largest problem with $d = 100$, despite the apparently poor models in terms of both AS recovery and predictive accuracy for the POD coefficients, the field level errors are consistently similar when compared to the lower-dimensional problems with $d = 25$ and $d = 50$ for sufficiently large n. This yet again corroborates the fact that when the spectrum of the snapshot matrix decays rapidly, the later modes' contribution to the field output becomes progressively insignificant. Consequently, the penalty in accuracy incurred at the field level due to models with poor predictive performance for the corresponding coordinates diminishes for the trailing POD modes. Figure 104 shows a visualization of the various components of the field level error for a randomly picked validation point for the problem with $d = 100$. Note that all the errors are at least an order-of-magnitude smaller than the actual value of the field output. Visual inspection only reveals minor differences.

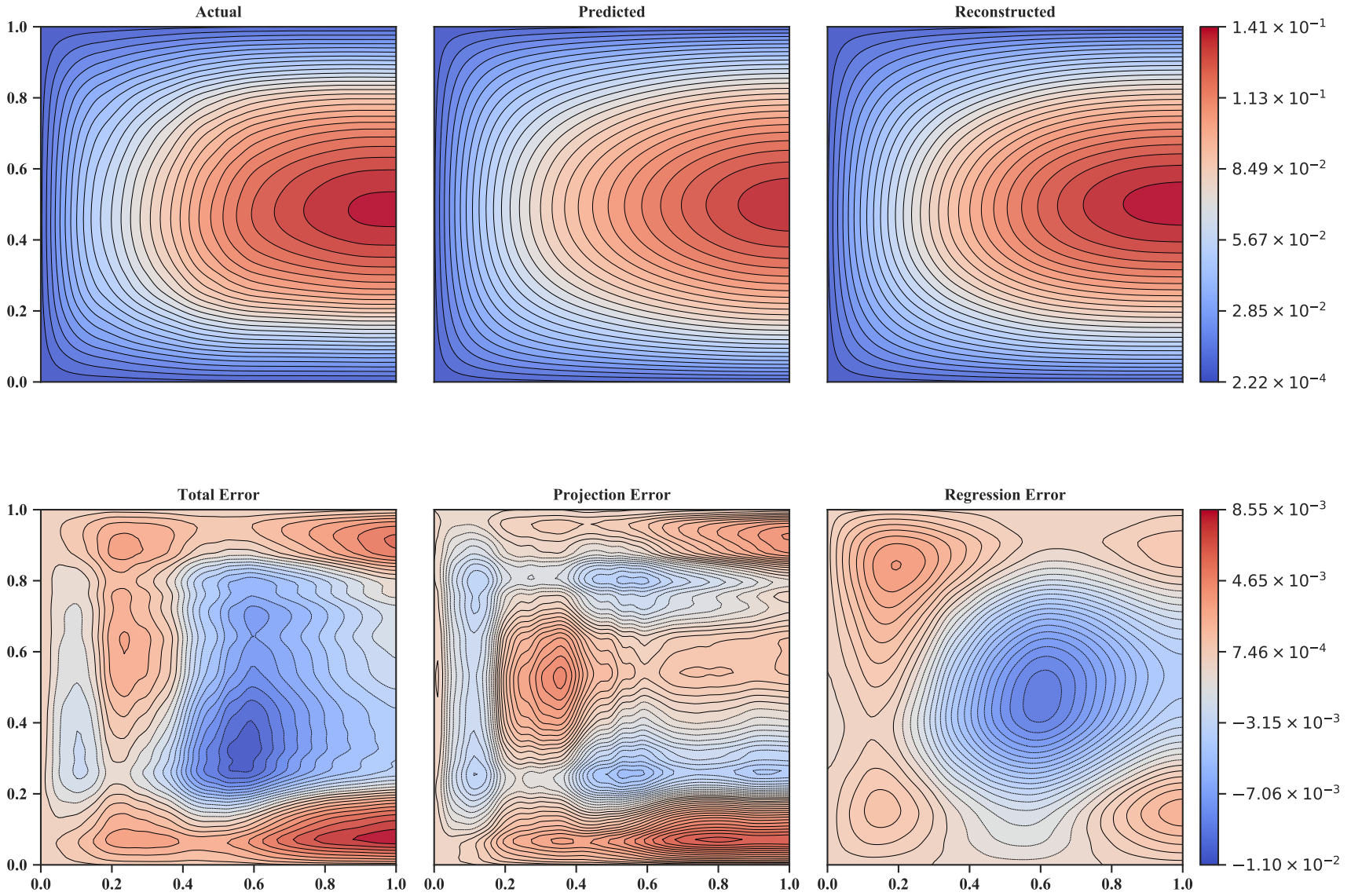


Figure 104: Visual Comparison for the Elliptic PDE Problem with $d = 100$ (Randomly Picked Test Point)

Results for the Airfoil Problems. Recall that we demonstrate the methodology on two subsonic airfoil flow problems with $d = 15$ and $d = 25$ in order to study the effect of increasing d for practical problems. The reader is referred to section 5.3.2.2 to get familiarized with the problem description before proceeding. Note that the airfoil problem represents a practical scenario for which the gradients of the POD coefficients are unavailable. Therefore, we are left with no choice but to solely rely on the BIC to check whether the AS exists, and if it does, set its appropriate size. It was noted from Figure 73 that an RIC of 99.9% is achieved when the truncation rank k for the problem with 15 and 25 input variables is 13 and 18, respectively. The minimum training dataset size to achieve small projection errors is 300 when $d = 15$ and 450 when $d = 25$. All the points not considered in training constitute the validation set.

Figure 105 and figure 106 show the variation of the BIC as q is increased for all the POD coefficients for the problems with $d = 15$ and $d = 25$, respectively. The results were obtained for 5 training repetitions with randomly sampled datasets. In contrast to the lower-rank elliptic PDE problem, the BIC for the dominant POD modes is relatively inconsistent, i.e. depends more on the points in the training set. Observe that similar to the elliptic PDE problem, on average the BIC values start peaking at higher q values as the POD coefficient index increases. Also note that for the coefficients corresponding to the trailing POD modes, the rise in the BIC becomes relatively more gradual, i.e. information contained in the noisier coordinates starts getting captured slowly as additional input dimensions are provided to the MO-GP model. Both figure 105 and figure 106 show two types of behaviors beyond the maximum BIC. The first kind is characterized by a marked decrease in the value of the BIC beyond its peak indicating that the improvement in the log-marginal likelihood is slower than the penalty incurred due to the additional model parameters. The second kind of behavior is characterized by a more-or-less constant BIC beyond the peak indicating that any improvement in the model is offset by the penalty incurred due to a model with increased complexity. In either case, a parsimonious model without too much loss in accuracy is typically desired.

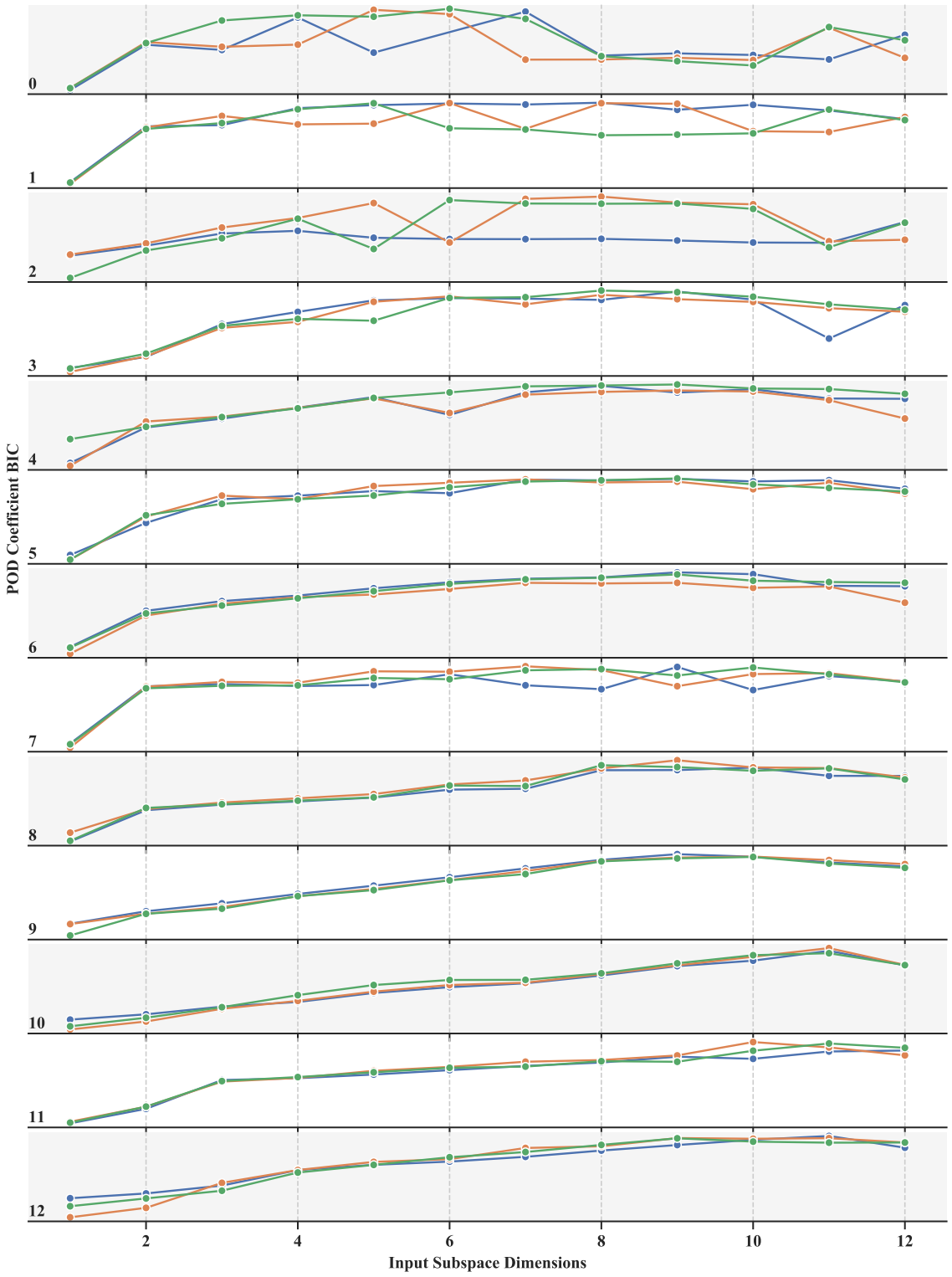


Figure 105: Variation of BIC for the 15 Variable Airfoil Problem. Colors Represent Different Repetitions.

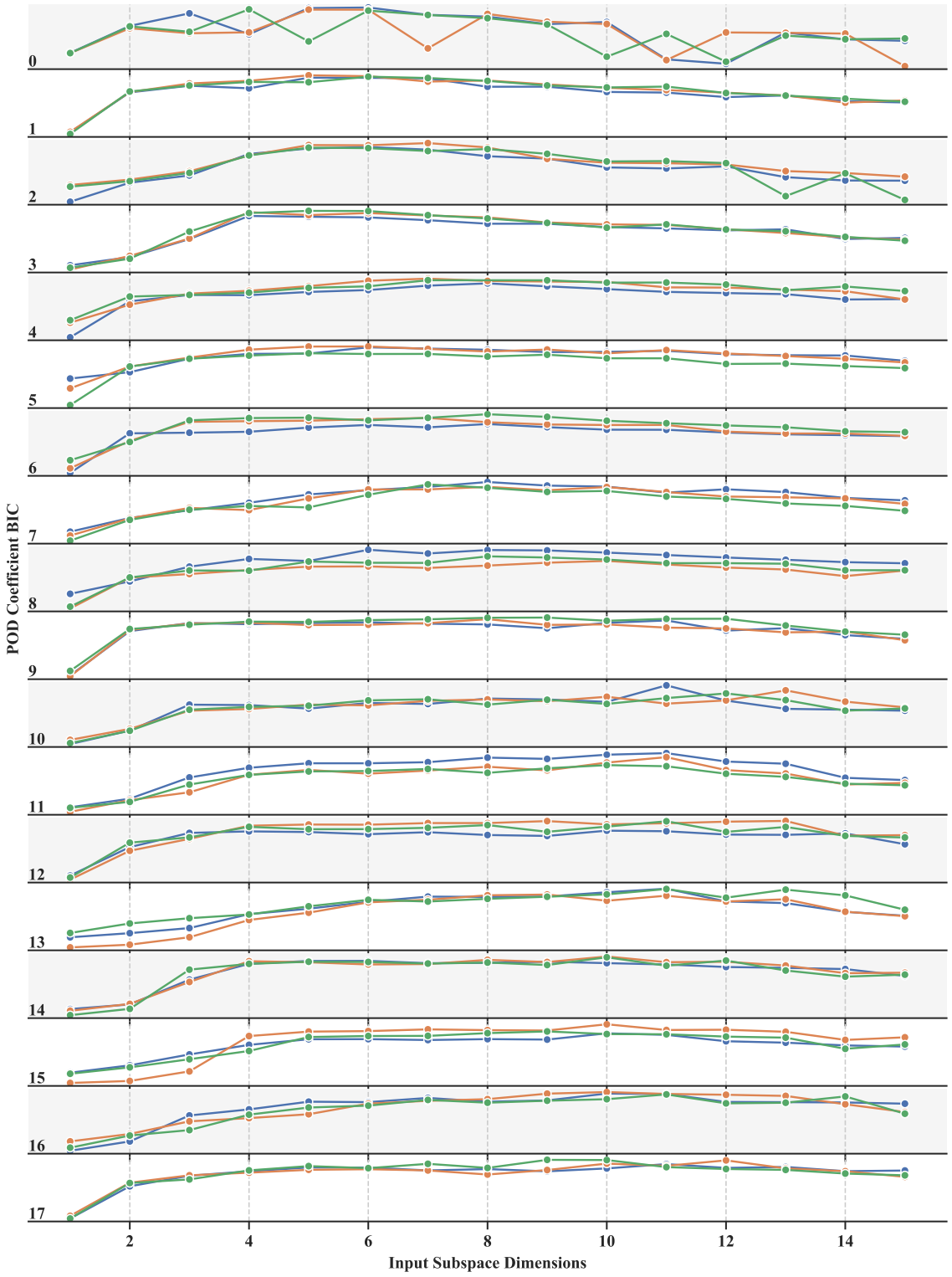


Figure 106: Variation of BIC for the 25 Variable Airfoil Problem. Colors Represent Different Repetitions.

Table 8: POD Coefficient Input Subspace Sizes for the Subsonic Airfoil Flow Problems

POD Coefficient	q	
	$d = 15$	$d = 25$
0	2	2
1	2	2
2	2	3
3	3	3
4	3	3
5	3	4
6	3	4
7	3	4
8	4	4
9	4	5
10	4	5
11	5	5
12	6	5
13	-	5
14	-	5
15	-	5
16	-	5
17	-	5

We proposed the use of V-NRMSE in addition to the BIC to ensure that preferring parsimony over exactly maximizing BIC does not result in an inaccurate model. Table 8 shows the values of q for each POD coefficient for all the airfoil problems obtained using both the behaviors of the V-NRMSE (not shown here for brevity) and the BIC. Notice that when $d = 25$, the onset of higher values for q begins earlier than it does for the problem with

$d = 15$, indicating the increase in problem complexity due to the finer shape parametrization. However, even the noisy trailing modes can be captured by at most $q = 0.33 \times d$ AS dimensions, i.e. there is strong evidence of the existence of a low-dimensional input structure for all the POD coefficients. The predictive performance of the MO-GP models resulting from our choice of q for each coefficient is shown in figure 107 and figure 108 for the 15 and 25 input variable airfoil problems, respectively. Similar to what was seen for the elliptic PDE problems, the trailing mode coefficient predictions become progressively worse because of their inherently noisy nature. Furthermore, for both problems with $d = 15$ and $d = 25$ the models for the POD coefficients yield more accurate and consistent V-NRMSE values when provided with a larger number of training samples. On closer inspection, it can be seen that the difference in the V-NRMSE values between models with differing sizes of training datasets increases for the trailing modes. This implies that larger training datasets benefit the predictive capacities of the MO-GP models for the relatively challenging-to-fit trailing coefficients more than they benefit the models corresponding to the dominant POD modes.

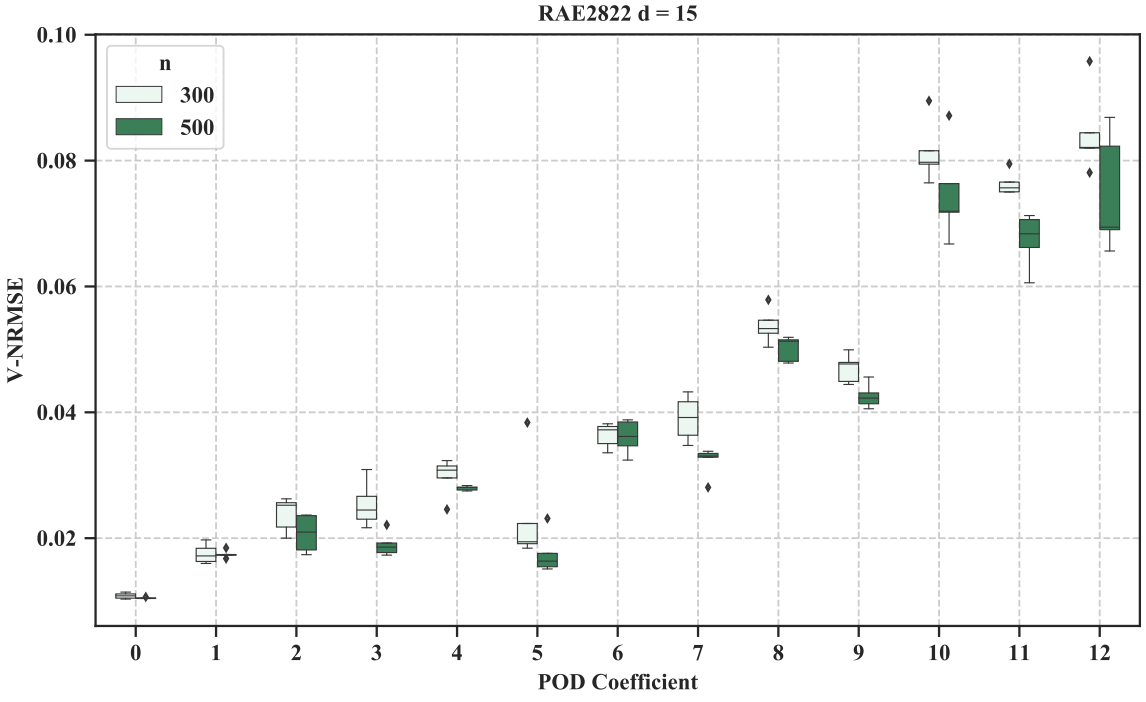


Figure 107: POD Coefficients' Validation NRMSE for the 15 Variable Airfoil Problem

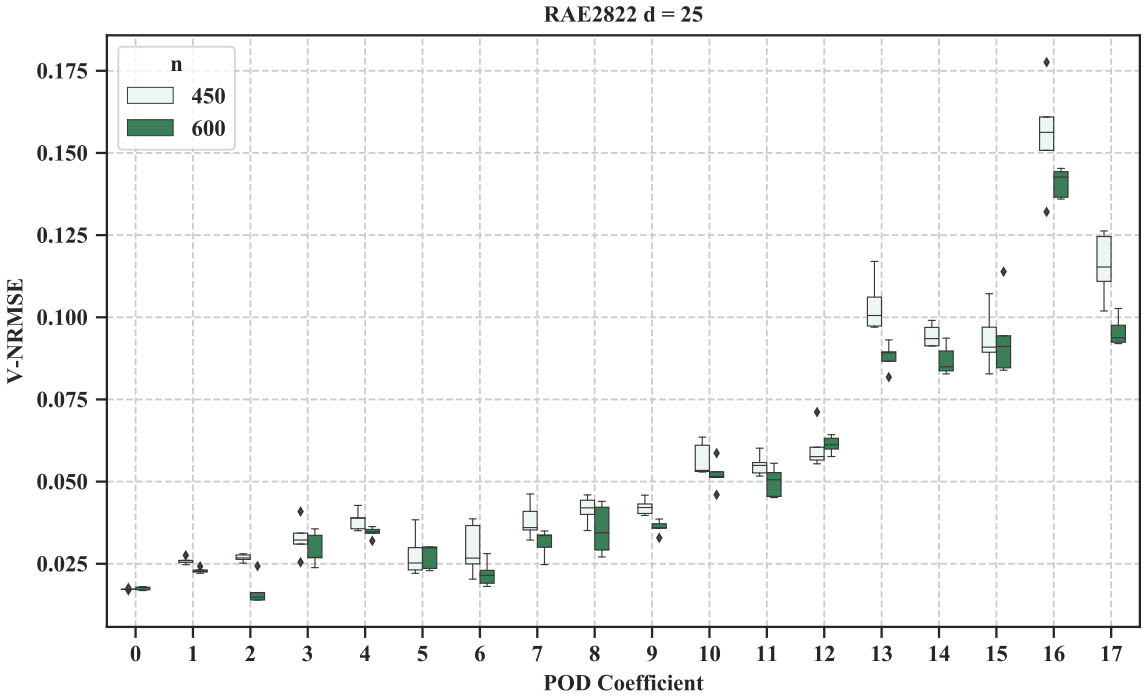


Figure 108: POD Coefficients' Validation NRMSE for the 25 Variable Airfoil Problem

The field level predictions for both the airfoil problems as obtained from 10 repetition using randomly sampled training datasets is shown in Figure 109 and Figure 110. As expected, having additional training samples results in similar but more consistent training regression and total errors. This behavior is more prominent for the problem with $d = 15$ than for the more challenging problem with $d = 25$. Note that these results are consistent with the ones for the elliptic PDE. Training and validation errors in the same order-of-magnitude indicate that the POD and interpolation models are not overfitting. More importantly, the effect of larger training datasets can be observed in the clear decrease in the field and regression errors for both the problems. In contrast to the case of the elliptic PDE problem, observe that the regression error is smaller than the projection error for the airfoil problems especially when n is large. This observation highlights the effectiveness of the MO-GP model.

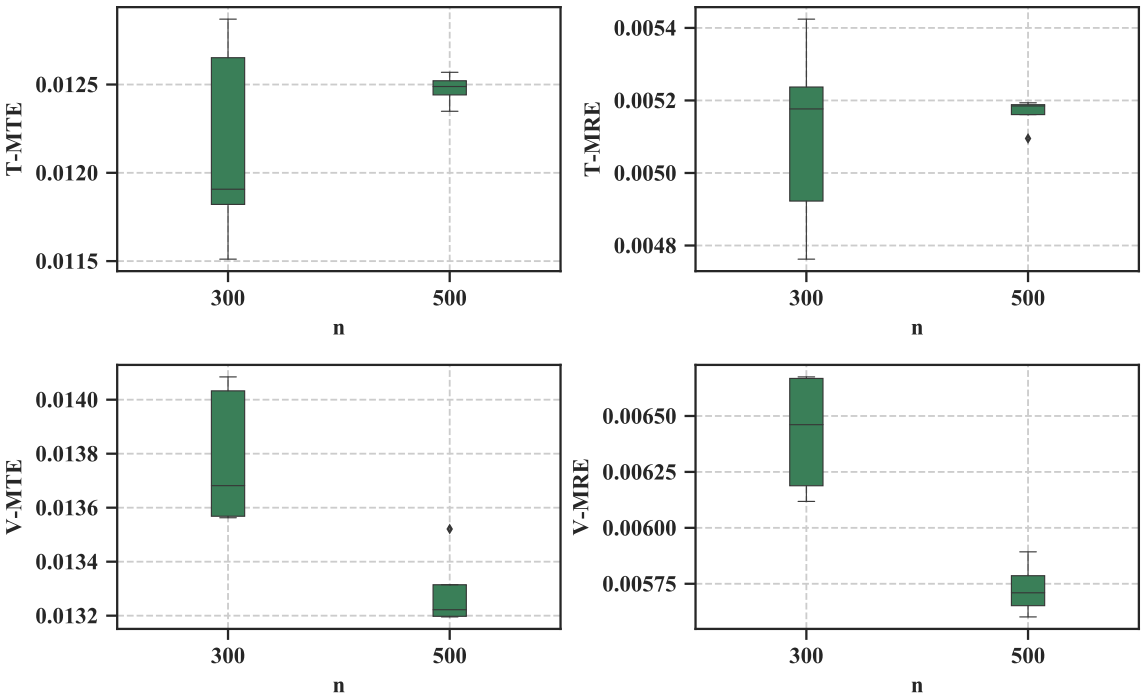


Figure 109: Field Error Metrics for the 15 Variable Airfoil Problem

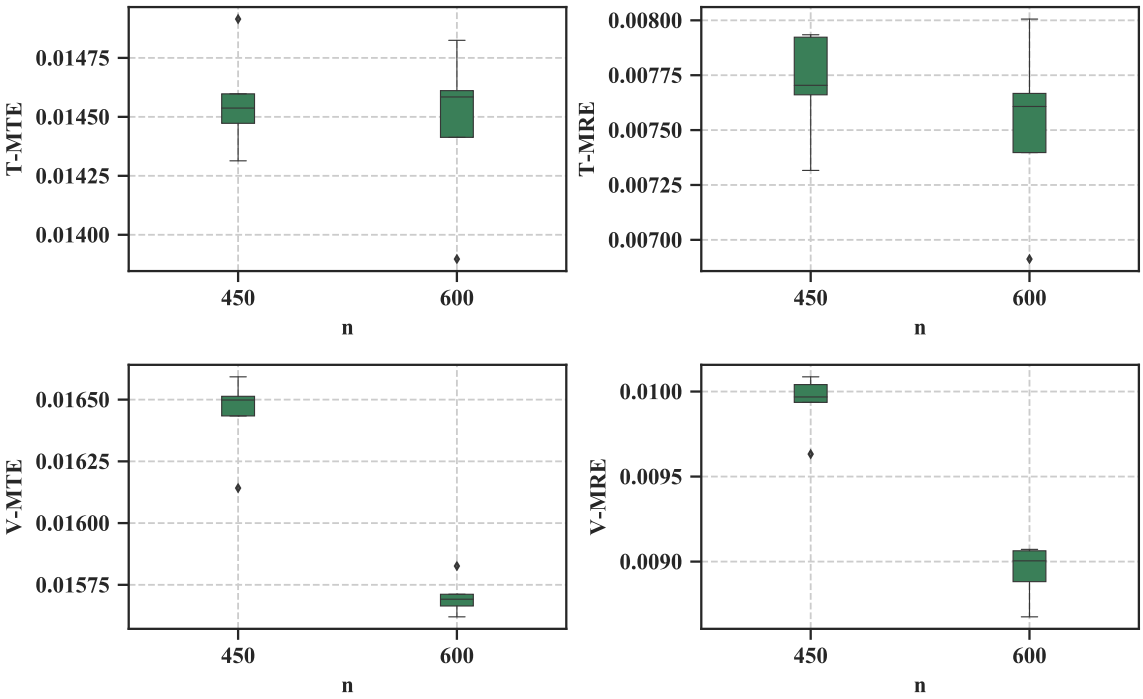


Figure 110: Field Error Metrics for the 25 Variable Airfoil Problem

Figure 111 shows the worst prediction made by the models trained for the 25 input variable airfoil problem to serve as a final visual check. Notice that the input values for this particular design lead to an inverted airfoil with suction on the bottom surface. Despite being the worst predicted point in the validation set, the majority of the point-wise errors consist of the lighter shades of blue close to zero.

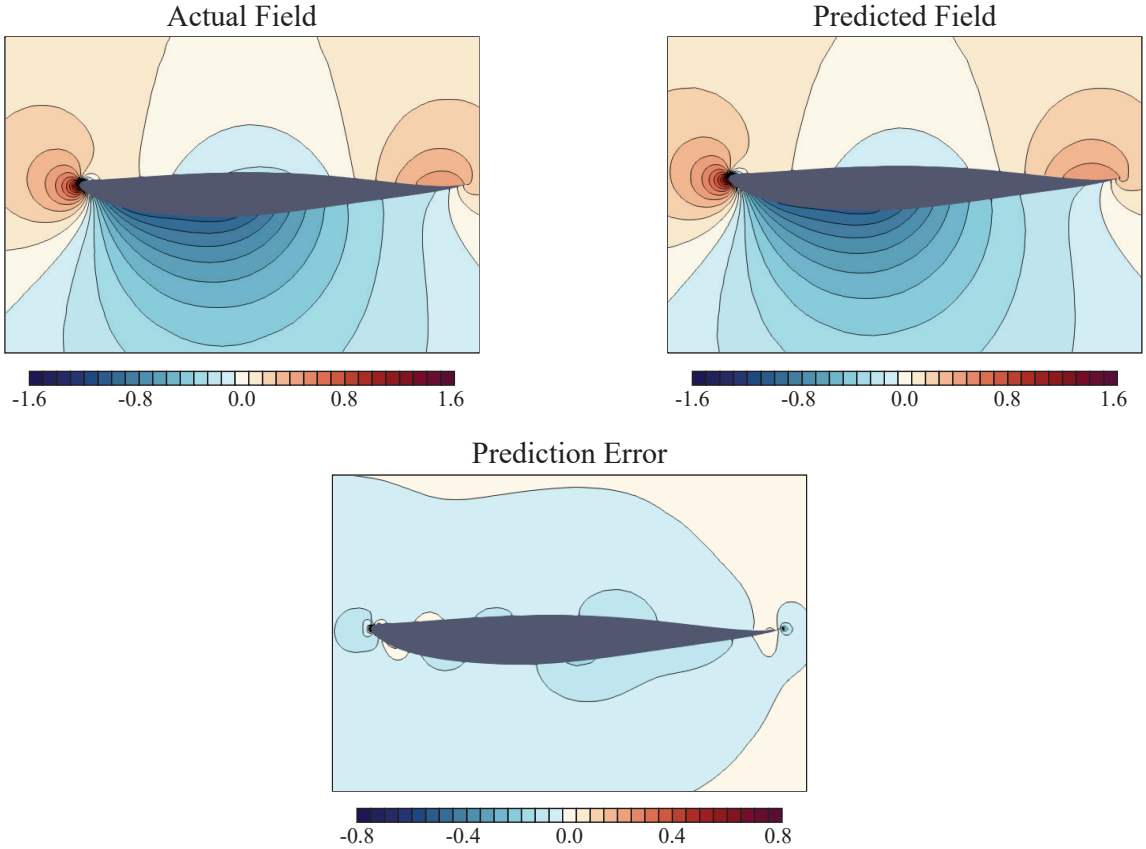


Figure 111: Visual Comparison of the 25 Variable Airfoil Problem for the Worst Prediction

Results for the CRM Wing Problem. The CRM wing with $d = 25$ shape variables is the most complex field emulation problem considered for demonstration under this research area. One, it is a distributed quantity in 3 spatial dimensions. Two, its parametrization is high-dimensional when compared to all the practical POD and interpolation-based ROMs studies in literature (see Table 2). Three, among all the test problems, the CRM wing exhibits the weakest decline in the singular value spectrum of its snapshot matrix. In fact, its decline is also relatively slower in the sense that as the trailing modes are approached,

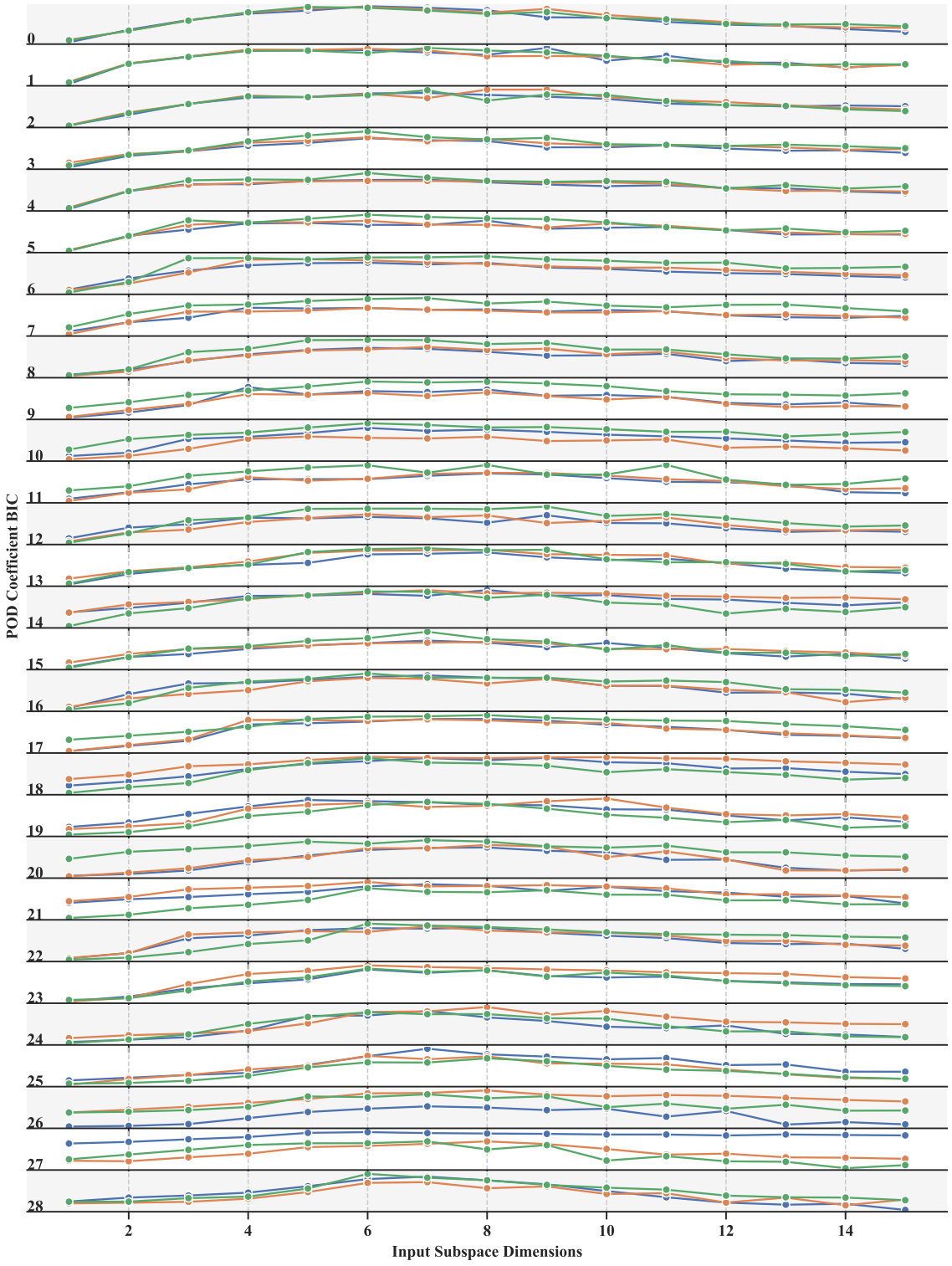


Figure 112: Variation of BIC for the 24 Variable CRM Wing Problem. Colors Represent Different Repetitions.

the rate at which the RIC accumulates decreases, i.e. the trailing modes are significant and must be predicted accurately. The reader is referred to section 5.3.2.3 before proceeding. Because of its complex, feature rich fields, training datasets containing 500 points (half of the size of the dataset) resulted in small projection errors.

The BIC peaks at relatively larger values of q for each POD coefficient when compared with both the airfoil and the elliptic PDE test cases, as shown in figure 112. This observation is a clear statement about the complexity of the CRM wing problem. Similar to the other test problems, observe that the BIC starts favoring larger values of q for the MO-GP models corresponding to the trailing, noisier POD modes. An important observation that differentiates the CRM wing problem form the previous problems is that the BIC seems to slowly rise to its maximum value at $q = 1/4 \times d = 6$ for all the POD coefficients. Fortunately, by using both the V-NRMSE (not shown here for brevity) and the BIC, we indeed discover the existence of low-dimensional input subspaces for all the 29 POD modes, as shown in table 9. Note that the size of the AS increases to $1/4 \times d$ within the first 15 POD modes beginning from a relatively high yet modest value of $1/8 \times d$.

Table 9: POD Coefficient Input Subspace Sizes for the Subsonic CRM Wing Problem

POD Coefficient	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
q	3	3	3	4	4	4	4	4	4	4	5	5	6	6	6
POD Coefficient	15	16	17	18	19	20	21	22	23	24	25	26	27	28	-
q	6	6	6	6	6	6	6	6	6	6	6	6	6	6	-

Figure 113 shows the predictive accuracy of the MO-GP models trained using q values reported in table 9. We present results obtained from 10 repetitions for each setting of n using randomly sampled training sets. The remaining points are used to construct the validation set. While the MO-GP models corresponding to the initial dominant POD modes

are accurate for both values of n, models corresponding to the mode indices 10-15 appear to be challenging to train, as indicated by the inconsistent and relatively poor predictive performance. On further investigation, it was found that this behavior could not be improved despite increasing the value of q. Increasing n_{restarts} could be another potential way of improving the models in question. However, we did not pursue this in order to maintain uniformity in the training algorithm settings for all the models. But for the relatively problematic models corresponding to the mid-range mode indices, we observe that it becomes progressively difficult to predict the trailing modes' coefficients.

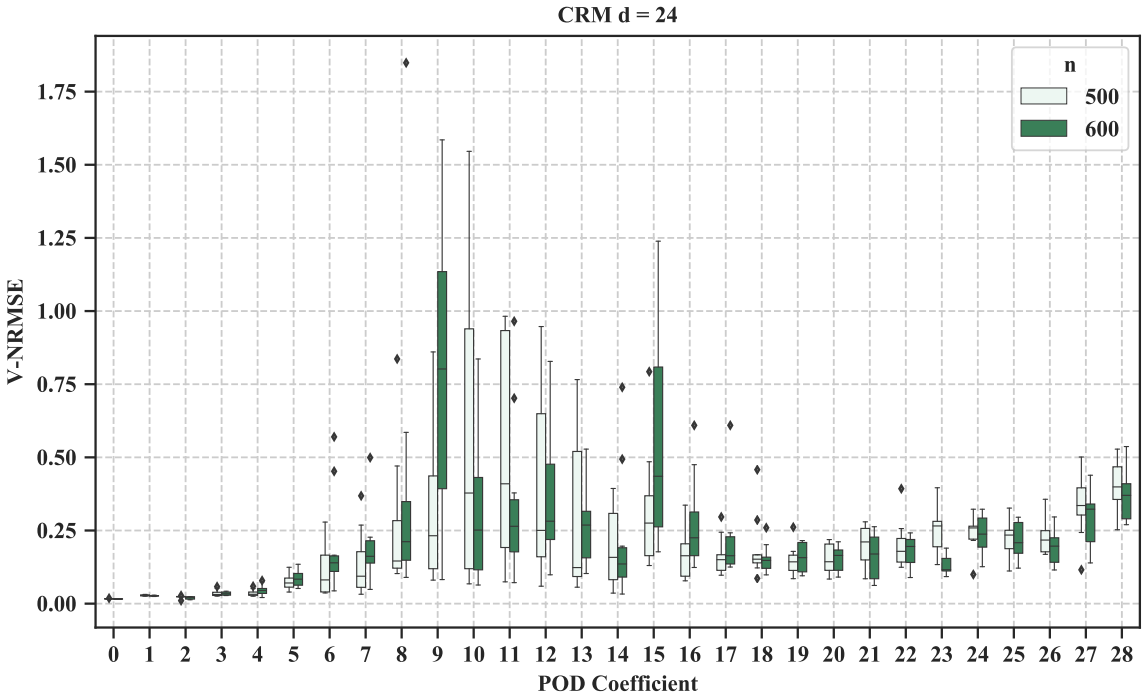


Figure 113: POD Coefficients' Validation NRMSE for the 24 Variable CRM Wing Problem

Despite the apparently poor MO-GP models for the coefficients in the mid-range, figure 114 shows the small errors achieved by the proposed methodology at the field level. Although the training error increases marginally when n is increased, both the total and regression error in the validation sets lie between $1 - 1.4\%$ and $0.8 - 1.2\%$, respectively. In line with the spirit of the airfoil problem and for the sake of transparency, we finally present a visual comparison of the worst prediction for the CRM wing problem in Figure 115.

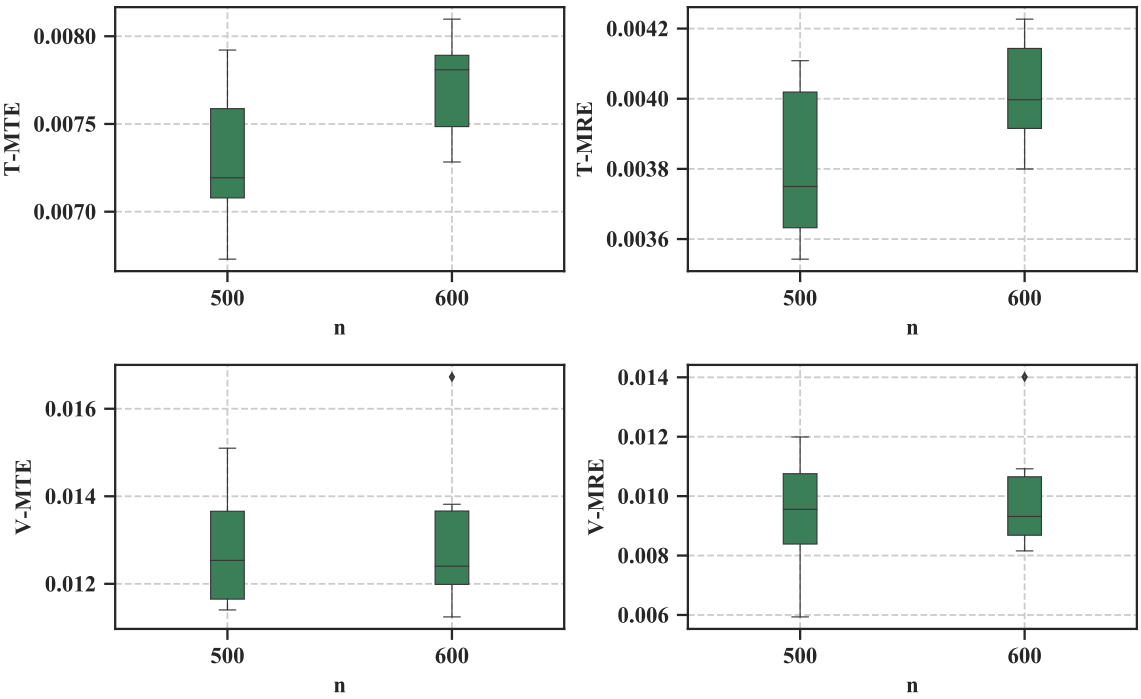


Figure 114: Field Error Metrics for the 24 Variable CRM Wing Problem

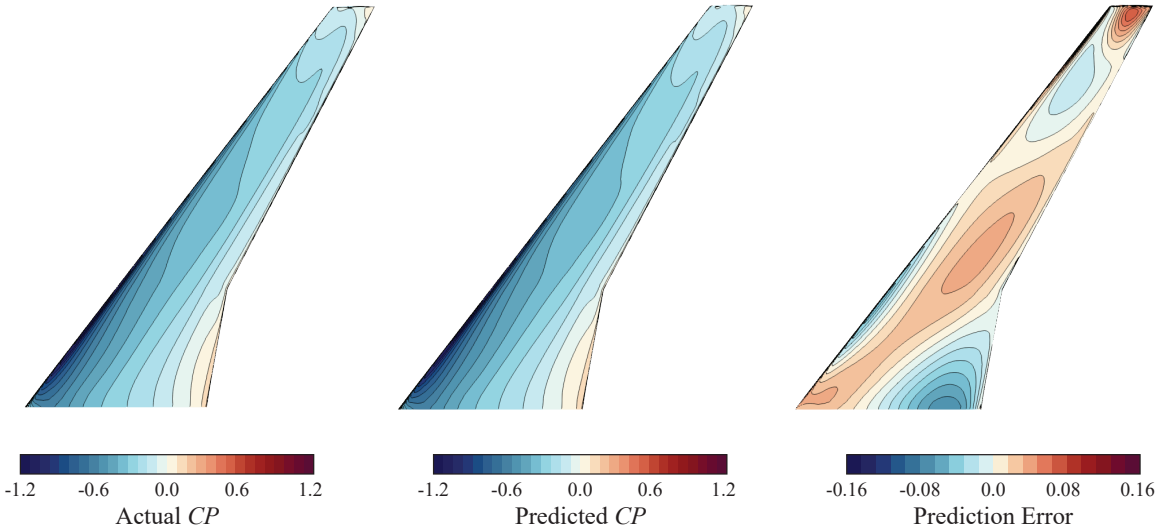


Figure 115: Visual Comparison of the 24 Variable CRM Wing Problem for the Worst Prediction

5.5 Summary and Concluding Remarks

This chapter proposed a solution to tackle the challenges posed by large input spaces in the construction of predictive ROMs. Since the interpolation/regression step is directly affected by large input spaces, we argued that any advancement must aim at enabling

supervised learning in high-dimensions. A thorough literature search put forth mapping-based strategies as the most suitable candidates to address the curse of dimensionality in the context of surrogate modeling in high-dimensions. Of specific interest were the methods that find a lower dimensional input subspace and perform the supervised learning in the lower dimensional subspace. The recently proposed ASM was identified as a method that has enjoyed immense success in practice. However, its reliance on gradients rendered it incompatible for our use case of nonintrusive ROMs. Other competing methods that approximated the gradients using point samples were found to rely on a structured sampling plan in order to provide a guaranteed successful recovery of the AS. Additionally, they required a large number of samples. We argued that any competing method must be able to work with a set of pre-tabulated input-output pairs.

The need for a supervised learning method that does not rely either on availability of gradients or on a structured sampling plan led to the proposal of a new manifold optimization-based Gaussian process regression model called the MO-GP. The model involved redefining the kernel on low-dimensional inputs obtained by projection of the original high-dimensional inputs onto a subspace parametrized using the Grassmann manifold. Both the parameters of the kernel for the GP and the projection subspace were computed simultaneously by maximizing the log-marginal likelihood of observing the training data on a product manifold. The product manifold comprised of a Euclidean part (the GP hyperparameters) and the Grassmann manifold (the projection subspace).

The MO-GP model's performance was analyzed and compared against a competing benchmark model called the GP-AS using a large suite of test functions. In general, the results showed that the MO-GP either outperformed or was at least competitive with the GP-AS method. Surprisingly, it was found that the MO-GP model successfully recovered the AS without requiring the gradients. However, the MO-GP required specification of the size of the input subspace a priori. This requirement made its use impractical. The choice of an appropriately sized subspace was posed as an exercise in model selection. We asserted and demonstrated that the BIC and the NRMSE are effective indicators for finding the size of low-dimensional input subspace that balances predictive accuracy and model complexity.

Following the tests on scalar functions, we proceeded with the use of MO-GP to learn the map between the POD coefficients and high-dimensional inputs by discovering an independent subspace for each coefficient. Using high-dimensional elliptic PDE problems of different sizes with known gradients obtained intrusively, we demonstrated that the MO-GP could successfully find a low-dimensional subspace for each POD coefficient. In general, the POD and interpolation-based ROMs using MO-GPs were successfully able to predict the field for unseen inputs in high-dimensions with small errors.

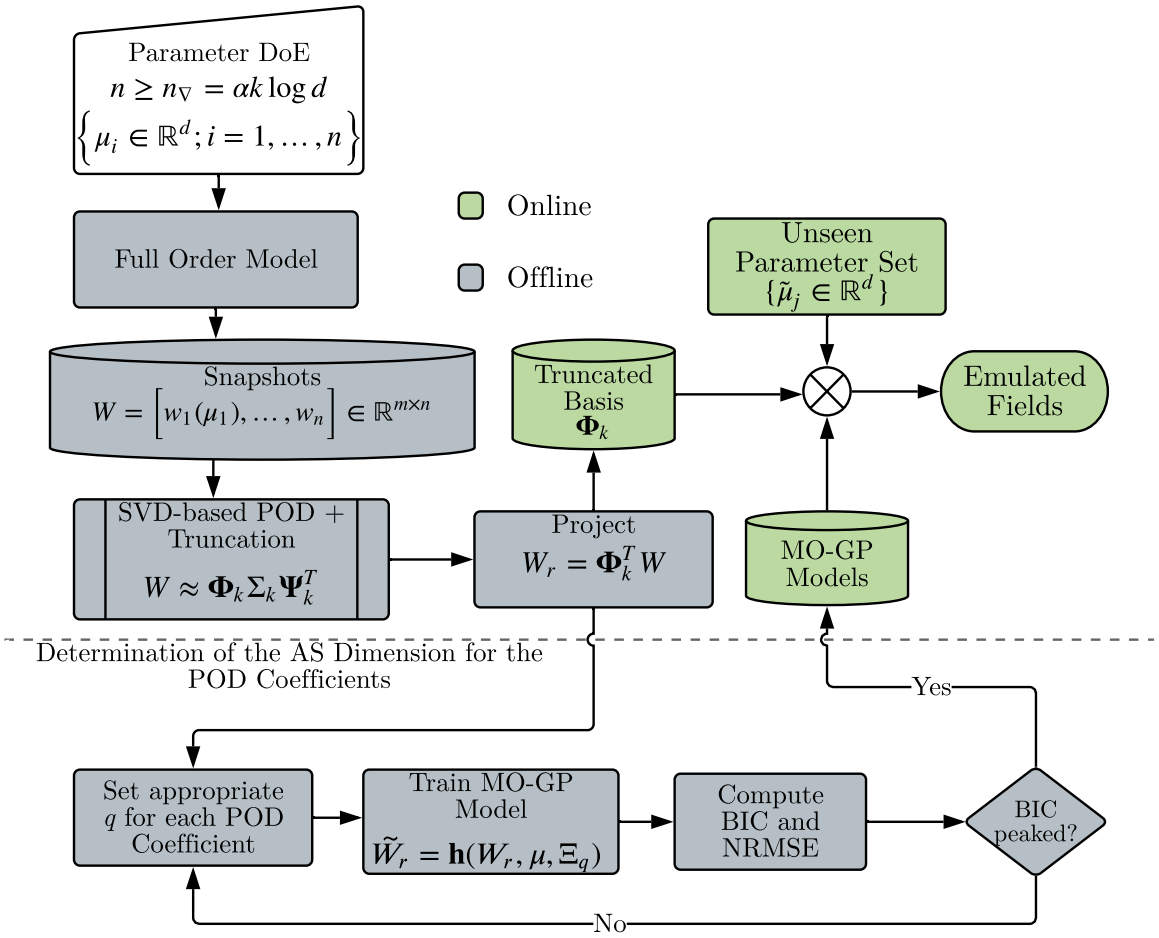


Figure 116: Offline-Online Decomposition for the POD and Interpolation-Based ROM using the MO-GP

Figure 116 summarizes the methodology. Before concluding, we provide some recommendations for the use of MO-GP to construct POD and interpolation-based ROMs based on our observations. Demonstration on all the test problems revealed the existence of a

small AS for the initial dominant POD modes. For the later and more noisy modes, the dimensionality of the AS increased significantly. In fact, the dimensionality of the AS progressively increased for higher mode indices. This observation highlights a cautionary note regarding the proposed methodology's limitation for certain kinds of problems. For some problems where the snapshot matrix displays a weak singular value spectrum, one can easily imagine a situation where the truncation rank is large enough such that the size of the AS corresponding to the trailing POD modes starts becoming comparable to the original size of the inputs. In other words, the variation of the coefficients for the trailing POD modes cannot be captured in low-dimensional subspaces for such problems. The proposed method will clearly fail to perform in such cases because of the inability of GPs to work in high-dimensions. On the other hand, there may also be problems with a slowly decaying singular value spectrum, variation of whose trailing POD modes can be accurately captured in a low-dimensional subspace using the MO-GP. The proposed methodology is precisely designed to work effectively in such situations.

CHAPTER VI

MANIFOLD ALIGNMENT-BASED MULTIFIDELITY POD AND INTERPOLATION USING GAUSSIAN PROCESS REGRESSION ON GRASSMANN MANIFOLDS

The work presented in this chapter attempts to address the challenges discussed under research area 3 in chapter 3. Figure 117 shows an overview of the key issue along with an outline of the order of presentation this chapter follows. First, we go through a brief

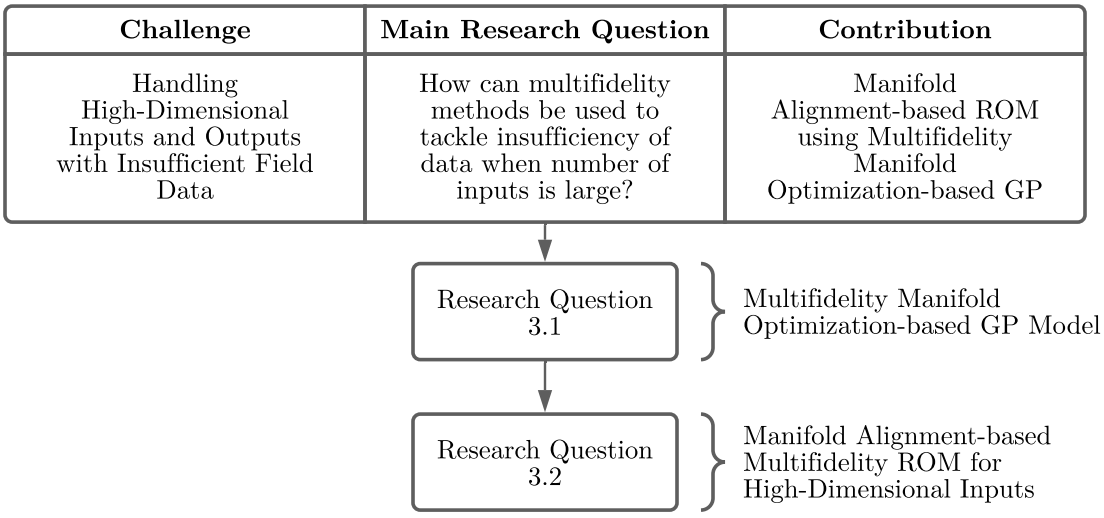


Figure 117: Chapter Outline

literature review of the techniques that have attempted to solve the challenges posed by the trade-off between the need for high accuracy and the high expense of generating training data, i.e., the need to achieve good predictive accuracy under limited high-fidelity data. Then, we show that multifidelity methods have the potential to provide a solution to the problem. However, the current state-of-the art for predicting fields using multiple sources of information has given limited attention to the problem of inconsistent fields between the datasets. Moreover, when the realistic requirement for the need to handle large parameter spaces is tagged on, there currently exists no method that can construct a nonintrusive

predictive ROM using input-output pairs from multiple datasets in high-dimensional input spaces.

So far, we have looked at managing the contribution of the dimension reduction step to the *offline* cost by leveraging randomization (see chapter 4). Let us now shift our focus to the other significant contributor i.e. the cost of generating high-fidelity data. Recall that we recognized the promise of multifidelity methods to tackle this challenge. When considered in the context of emulating field outputs for practical problems, any multifidelity method must provide solutions to the following (explained in detail in the subsequent sections) aspects:

1. Common low-dimensional representation of field outputs of potentially disparate sizes (or dimensionality) from different fidelities of analyses
2. Regression/Interpolation in the common low-dimensional representation
3. High-dimensional input parametrization
4. Input parametrizations of different sizes between the various fidelity levels

This dissertation focuses on the first three issues with a special focus on high-dimensional input parametrizations. We assume that the different fidelitites have the same design parametrization but have field outputs of different sizes and topologies. In fact, our test case is an example where the fields have inconsistent topologies. Before developing the proposed method, we set the stage for the need for a new approach through a brief literature search. Finally, we formulate a new approach that leverages information from different sources varying in their fidelity to construct multifidelity parametric ROMs in high-dimensional input spaces. The approach utilizes a newly proposed [195] manifold alignment-based approach to find a common low-dimensional representation of the field outputs and formulates a multifidelity extension of the MO-GP model developed in chapter 5 to train the surrogate model that captures the relationship between the latent space coordinates and the input parameters.

Remark. The number of repetitions for all the experiments in this chapter was set considering the available computational resources and to reveal any issues regarding bias

and/or inconsistency in convergence. Beyond the reported number of repetitions for each experiment, the distribution of the results did not significantly change the observations and conclusions.

6.1 Multifidelity Regression and ROMs

Multifidelity methods for both data-fit scalar surrogate models and ROMs can be divided into *adaptation*- and *fusion*-based methods. As its name suggests, the former class of methods corrects the low-fidelity results to match the high-fidelity results using an appropriate correction function [6, 67, 244]. On the other hand, the latter class of methods unifies information from varying levels of fidelity into a single synergistic model. Fusion-based methods are relatively more accurate than adaptation-based methods [114, 188]. However, their training procedure is relatively involved and their implementation is challenging when compared with adaptation-based methods. In recognition of their higher accuracy, we exclusively focus on fusion-based methods going forward. One of the most famous fusion-based multifidelity data-fit model goes by the name of CoKriging [95] or multifidelity Gaussian Process Regression [140]. These models essentially use individual GPR models for the low- and high-fidelity data and encode the relationship between the data sources by modifying the covariance functions. A classical example is the autoregressive formulation of Kennedy and O'Hagan [129] especially suited for computer experiments. Because our previous efforts in chapter 5 focused on GP-based surrogate model to regress the data in the reduced representation of the field outputs, we focus on the multifidelity GPR model to attempt the formulation of an extension to the MO-GP method to handle a large number of inputs. Furthermore, all the advantages of adopting the GPR model for use with the mapping-based strategy to handle high-dimensional inputs apply here as well.

Let us now shift our focus on multifidelity ROMs. Under the *adaptation* kind, Malouin et. al. [164] propose combining solutions from a fine and a coarse grid solving the same governing equation. The authors [164] then proceeded with the construction of a ROM for the difference between the high- and interpolated low-fidelity field solutions. The solution for an unseen parameter was obtained by evaluating the simulation on the coarse grid and

adding to it the error predicted by the ROM. In this sense, the error ROM plays the role of an additive correction.

Toal [246] proposed a multifidelity ROM based on the Gappy POD [91]. Their method relies on the Gappy POD method to impute the values of the high-fidelity solutions using low-fidelity data. In other words, the low- and high-fidelity solutions are stacked into a tall and thin matrix with some missing entries corresponding to the high-fidelity solutions. A least-squares regression problem is then solved to fill-in the missing entries. In this sense, the method treats the low-fidelity solutions as inputs to the predictive model

Another very recent proposal [259] constructs a map between the reduced representation (obtained by independently performing the POD) of the low- and high-fidelity data. The high-fidelity solutions at unseen input values are predicted by solving the low-fidelity simulation, projecting onto the corresponding POD basis, evaluating the trained map to obtain the projected coordinates of the high-fidelity analysis, and finally using high-fidelity POD basis to reconstruct the solution.

The ROM methods under the *fusion* kind can be partitioned into methods that fuse data into a common POD basis and methods that intelligently fuse the compressed representation of the high- and low-fidelity data. Researchers [37,175] have demonstrated the effectiveness of working with a common POD basis. The basic approach involves obtaining a snapshot matrix containing both low- and high-fidelity solutions. Note that the authors recommend that any inconsistency in the size of the outputs be handled using spatial interpolation. The POD is performed on this common snapshot matrix. Parametric dependence is encoded by treating the projected coordinates for the low- and high-fidelity as a multifidelity dataset. Any of the previously described multifidelity scalar surrogate models may be invoked to train a regression model for the multifidelity dataset.

Instead of finding a common POD basis which inherently favors the dataset with abundant samples, Benamara et. al. [27] propose extending the POD basis of the high-fidelity dataset with that of the low-fidelity dataset. In order to avoid capture of redundant information, their method projects the low-fidelity data onto the null space of the high-fidelity dataset. The resulting extended basis is then used to project both the high- and low-fidelity

solutions to obtain their respective POD subspace coordinates. Finally, a multifidelity supervised learning model is trained on the obtained low-dimensional representation.

Before proposing the use of a newly developed multifidelity ROM for our purpose, we now discuss some pros and cons of the existing methods presented above. For all the multifidelity ROM methods, the issue of inconsistent field outputs is resolved by interpolating onto a common grid. However, they completely ignore cases where the low-fidelity data has different features or comes from a grid with a different topology. In such cases, the interpolation onto a common grid is ill-defined, if at all possible. For methods that work with inconsistent fields without modification, observe that the prediction at an unseen point requires evaluation of the low-fidelity analysis which may be considerably expensive. Addressing these limitations is the main motivation behind a recently proposed [195] *fusion*-based multifidelity ROM that relies on manifold alignment and multifidelity regression. Therefore, it is our choice for tackling the issue of insufficient high-fidelity data for analyses with a large number of inputs. We now present its details and develop an extension to it for handling high-dimensional inputs.

6.1.1 Manifold Alignment-Based POD and Interpolation ROM

Consider the snapshot matrices $\boldsymbol{W}_l \in \mathbb{R}^{m_l \times n_l}$ and $\boldsymbol{W}_h \in \mathbb{R}^{m_h \times n_h}$ containing low- and high-fidelity datasets, respectively. Manifold alignment assumes that even though the columns in $\boldsymbol{W}_l$ and $\boldsymbol{W}_h$ may be of different sizes and may span different subspaces, i.e. $m_l \neq m_h$, they share a common representation in some low-dimensional latent space. Its primary purpose [113, 256] is to uncover such a shared latent space. Perron et. al. [195] recognized that even though $\boldsymbol{W}_l$ and $\boldsymbol{W}_h$ are results from equivalent design problems, it is reasonable to assume that they both share a common low-dimensional latent space in which the embedded data can be compared due to their intrinsic similarity.

We focus on the first kind of manifold alignment proposed by [256] that allows the use of any kind of dimension reduction method to first find their individual latent spaces. This particular type is of special interest for our work because the POD can be leveraged to find

the individual latent spaces in-line with all the work done in this dissertation. The subsequent alignment of the latent spaces thus obtained is done using the so-called Procrustes analysis. We just provide the steps for the resulting ROM method. For details, the reader is referred to [195].

Let $\boldsymbol{W}_l \in \mathbb{R}^{m_l \times n_l}$ and $\boldsymbol{W}_h \in \mathbb{R}^{m_h \times n_h}$ denote the snapshot matrices of the low- and high-fidelity analysis respectively. It is further assumed that $n_h < n_l$. Let the first n_h solutions be generated at common input parameter locations. The low-fidelity naturally partitions into $\boldsymbol{W}_l = [\boldsymbol{W}_{lc}, \boldsymbol{W}_{lu}]$. $\boldsymbol{W}_{lc} \in \mathbb{R}^{m_l \times n_h}$ contains the common (or linked) solutions and $\boldsymbol{W}_{lu} \in \mathbb{R}^{m_l \times (n_l - n_h)}$ contains the additional low-fidelity samples (i.e. unlinked data).

The method naturally partitions into an *offline* and an *online* phase. The offline phase consists of the following steps:

1. *Generate Linked and Unlinked Data:* Generate a large space-filling DoE. Evaluate both the high- and low-fidelity analyses at the first few common locations depending on the affordable computational budget. For the remaining points in the DoE, evaluate the inexpensive low-fidelity analysis. This first step leaves us with the snapshot matrices $\boldsymbol{W}_l$ and $\boldsymbol{W}_h$.

2. *Compute the POD:* Given some desire RIC value, apply the POD to $\boldsymbol{W}_l$ and $\boldsymbol{W}_h$ to obtain their respective POD modes, i.e., $\boldsymbol{\Phi}_{lk}$ and $\boldsymbol{\Phi}_{hk}$, as well as their latent variables $\boldsymbol{W}_l^r$ and $\boldsymbol{W}_h^r$. At this point, $\boldsymbol{\Phi}_{lk}$ can be discarded because we only require predictions for the high-fidelity analysis.

3. *Align Manifolds:* Apply Procrustes manifold alignment to find the optimal transformation of $\boldsymbol{W}_l^r$ into $\boldsymbol{W}_{l,\text{aligned}}^r$.

4. *Fit Regression Model:* Train the multi-fidelity regression models $\boldsymbol{h}(\boldsymbol{\mu}) = [h_1(\boldsymbol{\mu}), \ldots, h_k(\boldsymbol{\mu})]$ by combining both $\boldsymbol{W}_h^r$ and $\boldsymbol{W}_{l,\text{aligned}}^r$.

In the online phase, the prediction is performed similar to the POD and interpolation-based ROM:

1. *Predict POD Subspace Coordinates:* Given an unseen input $\boldsymbol{\mu}^*$, predict the corresponding latent space coordinates $\tilde{\boldsymbol{w}}_h^r$ by invoking the multifidelity regression model using $\boldsymbol{h}(\boldsymbol{\mu})$.

2. *Reconstruct High-Fidelity Field:* Using $\boldsymbol{\Phi}_{hk}$ and the POD ansatz, compute the predicted value of the unseen high-fidelity solution $\boldsymbol{w}_{\text{predicted}}$ given $\tilde{\boldsymbol{w}}_h^r$.

6.1.2 Extension of the MO-GP for Multi-Fidelity Regression

Even with the manifold alignment-based multifidelity non-intrusive ROM, new fields are predicted by estimating their coordinates in the latent space using some multifidelity regression model. Recall that regression models to predict the latent space coordinates must also have a multi-fidelity formulation since we wish to leverage both the high- and the aligned low-fidelity latent variables $\boldsymbol{W}_h^r$ into $\boldsymbol{W}_{l,\text{aligned}}^r$, respectively. For their multi-fidelity ROM, Perron et. al. [195] used the hierarchical kriging [114] regression model to train each $h_i, \forall i = 1, 2, \ldots, k$. Hierarchical kriging is relatively simpler to implement and it offers competitive accuracy when compared with the cokriging model. Note however that the *curse of dimensionality* affects hierarchical kriging or any multifidelity GP-based model in the same way as it affects any distance-based surrogate model. Since this dissertation concerns itself with problems in large input dimensions, we must develop a multifidelity regression model that accommodates high-dimensional inputs.

Our proposal involves formulating the MO-GP model (see section 5.1.3) as a multifidelity GP regression problem using a common input projection subspace shared between both the high- and low-fidelity functions. As with the MO-GP model, our model proceeds with training a regular multifidelity GP using kernel functions defined in the common projected low-dimensional subspace. We use the autoregressive formulation of Kennedy and O'Hagan [129] because of their suitability for computer experiments. Note that a similar idea has been investigated in the broadly applicable framework of Raissi et. al. [201] called Multifidelity Deep GPs. However, the goal in their work involved using a neural network to transform the original inputs before passing them as arguments to the kernel function.

The idea was to increase the ability of the trained multifidelity model to predict highly non-linear functions under availability of limited high-fidelity data. Our formulation specifically focuses on being able to handle high-dimensional inputs by finding a low-dimensional input subspace. Instead of a neural network, we project the original inputs onto a linear subspace parametrized using the Grassmann manifold. In this sense, the proposed model is different in its primary goal and unique in its formulation. However, the details of our model can be developed using the notation and broad framework put forth by Raissi et. al. [201] as shown below. Before proceeding, the reader is referred to section 5.1.3 to get familiarized with GP regression and the MO-GP model.

Following the notation in [201], let $\{\boldsymbol{\mu}_l, \boldsymbol{f}_l\}$ and $\{\boldsymbol{\mu}_h, \boldsymbol{f}_h\}$ denote the low- and high-fidelity datasets evaluated using the respective analyses at n_l and n_h locations with $n_l \geq n_h$. Note that we only discuss the realistic case where low-fidelity data is abundant when compared to high-fidelity data. Let us begin with the autoregressive model form [129] given by

$$f_h(\boldsymbol{\mu}) = \rho f_l(\boldsymbol{\mu}) + \delta_h(\boldsymbol{\mu}) \tag{117}$$

where $f_l(\boldsymbol{\mu})$ and $\delta_h(\boldsymbol{\mu})$ are independent GPs and ρ can be thought of as the scaling factor between the high- and low-fidelity functions. The two independent GPs have kernels κ_l and κ_h and can be written as

$$\begin{aligned} \delta_h(\boldsymbol{\mu}) &\sim \mathcal{GP}\left(0, \kappa_h\left(\boldsymbol{\mu}, \boldsymbol{\mu}'\right)\right) \\ f_l(\boldsymbol{\mu}) &\sim \mathcal{GP}\left(0, k_l\left(\boldsymbol{\mu}, \boldsymbol{\mu}'\right)\right) \end{aligned} \tag{118}$$

The autoregressive model form leads to the following GP form for $f_h(\boldsymbol{\mu})$

$$f_h(\boldsymbol{\mu}) \sim \mathcal{GP}\left(0, \rho^2 \kappa_l\left(\boldsymbol{\mu}, \boldsymbol{\mu}'\right) + \kappa_h\left(\boldsymbol{\mu}, \boldsymbol{\mu}'\right)\right) \tag{119}$$

This leads to the following structure for the multifidelity GP

$$\begin{bmatrix} f_l(\boldsymbol{\mu}) \\ f_h(\boldsymbol{\mu}) \end{bmatrix} \sim \mathcal{GP}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \kappa_l\left(\boldsymbol{\mu}, \boldsymbol{\mu}'\right) & \rho\kappa_l\left(\boldsymbol{\mu}, \boldsymbol{\mu}'\right) \\ \rho\kappa_l\left(\boldsymbol{\mu}, \boldsymbol{\mu}'\right) & \rho^2\kappa_l\left(\boldsymbol{\mu}, \boldsymbol{\mu}'\right) + \kappa_h\left(\boldsymbol{\mu}, \boldsymbol{\mu}'\right) \end{bmatrix}\right) \tag{120}$$

The multifidelity extension of the MO-GP can be easily understood by introducing the kernel defined on the low-dimensional input space obtained by projection onto a subspace

defined by $\boldsymbol{\Xi}$

$$\kappa_{(\cdot)}^{DR}(\boldsymbol{\mu}, \boldsymbol{\mu}'; \boldsymbol{\Xi}, \boldsymbol{\theta}') = \kappa_{(\cdot)}(\boldsymbol{\Xi}^T\boldsymbol{\mu}, \boldsymbol{\Xi}^T\boldsymbol{\mu}'; \boldsymbol{\theta}') \tag{121}$$

$$\kappa_{(\cdot)}^{DR} : \mathbb{R}^d \times \mathbb{R}^d \times Gr(q, \mathbb{R}^d) \times \mathbb{R}^l \to \mathbb{R} \tag{122}$$

$$\boldsymbol{\mu}, \boldsymbol{\mu}', \boldsymbol{\Xi}, \boldsymbol{\theta}' \to \kappa_{(\cdot)}^{DR}(\boldsymbol{\mu}, \boldsymbol{\mu}', \boldsymbol{\Xi}) \tag{123}$$

Note that $(\cdot)$ is meant to signify that $\boldsymbol{\Xi}$ is shared by the kernels of both fidelity levels. This leads to the following multifidelity structure for the MO-GP model

$$\begin{bmatrix} f_l(\boldsymbol{\mu}) \\ f_h(\boldsymbol{\mu}) \end{bmatrix} \sim \mathcal{GP}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \kappa_l^{DR}(\boldsymbol{\mu}, \boldsymbol{\mu}') & \rho\kappa_l^{DR}(\boldsymbol{\mu}, \boldsymbol{\mu}') \\ \rho\kappa_l^{DR}(\boldsymbol{\mu}, \boldsymbol{\mu}') & \rho^2\kappa_l^{DR}(\boldsymbol{\mu}, \boldsymbol{\mu}') + \kappa_h^{DR}(\boldsymbol{\mu}, \boldsymbol{\mu}') \end{bmatrix}\right) \tag{124}$$

The training to obtain MLE of the parameters proceeds in a fashion similar to that of the MO-GP model, the only major difference being the introduction of additional hyperparameters due to the presence of two kernels κ_l^{DR} and κ_h^{DR}. We solve the optimization of the log-marginal likelihood function using Algorithm 7.

Through the multifidelity model proposed above, it is expected that the presence of abundant low-fidelity data will help refine the knowledge of the low-dimensional input subspace of the high-fidelity function. Note however that the projection subspace $\boldsymbol{\Xi}$ is shared, i.e. assumed to be common between the functions varying in fidelity. Sharing the subspace can have serious implications when the low- and high-fidelity functions have wildly different low-dimensional input subspaces that capture the majority of their respective variations. This model can only be expected to work under the assumption that there is a strong correlation between the low- and high-fidelity function. We argue that this is a fair assumption to make for engineering functions where a difference in fidelities typically results in either an additive or multiplicative offset.

6.2 Experiments and Results

We are now ready to analyze the effectiveness of the combination of the manifold alignment-based ROM used in conjunction with the multifidelity MO-GP model. Of specific interest is to test the multifidelity MO-GP's ability to improve knowledge about the low-dimensional subspace given a large number of inexpensive low-fidelity data in addition to a few high-fidelity. Through the experiments performed, we also compare the multifidelity MO-GP

with the single-fidelity MO-GP to measure gains in predictive accuracy and improvements in the quality of the discovered subspace across a range of data availability scenarios.

The first set of tests are performed on scalar functions to gain a thorough understanding of the multifidelity MO-GP method's intricacies. Moreover, these modestly sized scalar functions lend themselves to easy modification and manageable training times. Using the insights gleaned from the tests on scalar functions, we finally assess its applicability with the manifold alignment-based ROM (MA-ROM) on a practical application problem.

6.2.1 Test Problem Description

We consider two engineering multifidelity scalar functions to perform the first experiment to whet the multifidelity MO-GP model. The first function is the 8-dimensional borehole function we used in chapter 5. The expressions for its high- and low-fidelity variants are

$$f_h(\boldsymbol{\mu}) = \frac{2\pi T_u \left(H_u - H_l\right)}{\ln\left(r/r_w\right)\left(1 + \frac{2LT_u}{\ln(r/r_w)r_w^2 K_w} + \frac{T_u}{T_l}\right)} \tag{125}$$

$$f_l(\boldsymbol{\mu}) = \frac{5T_u \left(H_u - H_l\right)}{\ln\left(r/r_w\right)\left(1.5 + \frac{2LT_u}{\ln(r/r_w)r_w^2 K_w} + \frac{T_u}{T_l}\right)} \tag{126}$$

where

Table 10: Description of Variables for the Borehole Function

Variable Name	Description
r_W	radius of borehole (m)
r	radius of influence (m)
T_u	transmissivity of upper aquifer (m^2/yr)
H_u	potentiometric head of upper aquifer (m)
T_1	transmissivity of lower aquifer (m^2/yr)
H_1	potentiometric head of lower aquifer (m)
L	length of borehole (m)
K_w	hydraulic conductivity of borehole (m/yr)

The description of the inputs only serve the purpose of giving an intuition. All experiments use scaled inputs such that they take values between -1 and 1. The second scalar function is a 10-dimensional nonlinear multifidelity variant of the wing weight function. Its high- and low-fidelity variants are given by

$$f_h(\boldsymbol{\mu}) = 0.036 S_w^{0.758} W_{fw}^{0.0035} \left(\frac{A}{\cos^2(\Lambda)}\right)^{0.6} q^{0.006} \lambda^{0.04} \left(\frac{100 t_c}{\cos(\Lambda)}\right)^{-0.3} (N_z W_{dg})^{0.49} + S_w W_p \tag{127}$$

$$f_l(\boldsymbol{\mu}) = 0.03 S_w^{0.758} W_{fw}^{0.0035} \left(\frac{A}{\cos^2(\Lambda)}\right)^{0.6} q^{0.005} \lambda^{0.04} \left(\frac{100 t_c}{\cos(\Lambda)}\right)^{-0.3} (N_z W_{dg})^{0.49} + S_w W_p + 20 \tag{128}$$

where

Table 11: Description of Variables for the Wing Weight Function

Variable Name	Description
$\mathrm{S_w}$	wing area $\left(\mathrm{ft}^2\right)$
$\mathrm{W_{fw}}$	weight of fuel in the wing (lb)
$\mathrm{A} \in [6, 10]$	aspect ratio
$\Lambda \in [-10, 10]$	quarter-chord sweep (degrees)
$\mathrm{q} \in [16, 45]$	dynamic pressure at cruise $\left(\mathrm{lb/ft}^2\right)$
$\lambda \in [0.5, 1]$	taper ratio
$\mathrm{t_c} \in [0.08, 0.18]$	aerofoil thickness to chord ratio
$\mathrm{N_z} \in [2.5, 6]$	ultimate load factor
$\mathrm{W_{dg}} \in [1700, 2500]$	flight design gross weight (lb)
$\mathrm{W_p} \in [0.025, 0.08]$	paint weight (lb/ft ^{2})

Note that both the borehole and wing weight functions have a nonlinear modification in their low-fidelity variant. Furthermore, after inspecting the spectrum of the approximation of $\boldsymbol{C}$ computed using $2,000$ gradient samples, we found that the low- and high-fidelity variants of both the functions have active subspaces of identical size. The trends of eigenvalue

decay also appear similar, at least visually.

For demonstration on a field emulation problem, we omit the detailed study done using the elliptic PDE for the sake of brevity. Moreover, as seen for the experiments in chapter 5, the behavior of MO-GP observed for scalars is a pretty good indicator of its behavior with field emulation problems. Instead, we focus on testing the proposed methodology on the 15 input variable subsonic airfoil flow problem detailed in section 5.3.2.2. The pressure coefficient distribution in the entire spatial domain is considered as the high-fidelity field output. Instead of creating another low-fidelity dataset and to simplify the first application, the coefficient of pressure restricted to the surface of the airfoil is considered as the low-fidelity output. Note that by construction, the multifidelity field dataset for our flow emulation problem consists of field solutions from inconsistent topologies. Such a situation is commonly encountered when panel methods-based aerodynamic codes are employed as low-fidelity solvers.

The main goal for this research area is to merely assess the feasibility and viability of the proposed multifidelity ROM. A detailed study, similar in spirit to the second research area is deferred to future work.

6.2.2 Effectiveness of the Multifidelity Manifold Optimization-Based GP Model

As a first step towards assessing the multifidelity MO-GP's ability to accomplish its primary goal of refining the quality of the AS using low-fidelity data, we posed the following research question:

Research Question 3.1

Does the proposed manifold optimization-based multifidelity model aid in improving the recovery of the low-dimensional input subspace and the predictive accuracy for the high-fidelity output using additional inexpensive low-fidelity data?

Recall that the MO-GP model recovered the true AS of the underlying function when given enough training samples. The ideal expectation from the multifidelity MO-GP model is that it finds a subspace that is identical to the true AS of the high-fidelity function

using a larger number of low- than high-fidelity samples. In the typical case of relatively abundant low-fidelity samples, this expectation is unfair and unreasonable given that the subspace is shared between the low- and high-fidelity models and only sparse high-fidelity data is available. In fact, it is fair to assume that the low-dimensional subspace will favor the true AS of the low-fidelity function. However, if the low- and high-fidelity functions are highly correlated in the sense that their variation in the domain is more or less similar, their respective active subspaces will also be similar. Under such cases, having a relatively large number of low-fidelity samples should make the multifidelity MO-GP find a subspace that resembles the one for the high-fidelity function. It is expected that the need to maximize the likelihood of observing the high-fidelity samples in the common subspace will help in its refinement to resemble the true AS of the high-fidelity function. Therefore, under the assumption that the functions varying in fidelity have similar active subspaces, it is reasonable to expect the following:

Hypothesis 3.1: *For a given function with a low-fidelity variant, if the number of high-fidelity samples is insufficient for accurately capturing the low dimensional input subspace, then additional low-fidelity samples will assist in a relatively accurate capture of the subspace and consequently result in a model with better predictive accuracy for the high-fidelity function.*

Knowing whether functions varying in fidelity have dissimilar active subspaces a priori is impossible unfortunately. The only rudimentary way of making an informed guess is to analyze outputs from multifidelity data evaluated at common input values. Instead of focusing on cases where we know the method will struggle, we use engineering functions that have analytical gradients. Before performing the experiment, it can be made sure that the multifidelity functions have a similar AS. In cases where gradients are unavailable, there is no choice but to assume that a common low-dimensional subspace can help refine the quality of the multifidelity MO-GP model and its estimate of the true AS of the high-fidelity function.

The basic idea for the experiment is to train several multifidelity MO-GP models for the

test functions and measure their ability to recover the AS and their predictive accuracy. As with all our previous experiments similar in nature, we employ the LPSA (eq. (52)) metric to measure discrepancy between the subspace computed by the multifidelity MO-GP and the true AS. The NRMSE (eq. (106)) metric computed on the validation set will serve as an indicator of the predictive capacity of the surrogate model for this experiment.

Details of the Experiment. Studies concerning multifidelity models must setup the experiment such that a fair comparison with the single-fidelity models is possible. Merely observing the performance of the multifidelity models does not highlight the extent of the benefit provided by the multifidelity techniques in comparison to their single-fidelity counterparts.

This experiment begins with the construction of a large space-filling DoE. Both the high- and low-fidelity functions are evaluated for the points in the DoE. Then, in accordance with the need to have linked data for the multifidelity ROM (which is the eventual application we are concerned with), we proceed with the construction of randomly sampled training sets with common input-output pairs for both the fidelity levels equal in number to the size of the high-fidelity training set (if any). The additional low-fidelity samples are randomly chosen from the remaining low-fidelity data. In order to make a fair comparison with the single-fidelity MO-GP model, a variety of multifidelity training sets are constructed with different ratios of the number of low- to high-fidelity training samples. Note that these different ratios represent disparate affordable budgets and computational cost discrepancies between the computer codes. Models with ratios of $0, 2, 5,$ and 10 are trained for different levels of the affordable number of high-fidelity samples. A ratio of 0 indicates that the training set consists of high-fidelity data only and the single fidelity MO-GP model is called upon for the training process. Finally, the trained models are used to compute the LPSA of the computed subspace with the true AS of the high-fidelity function obtained using $2,000$ gradient samples. Additionally, the V-NRMSE is computed on the validation points for predictions of the high-fidelity function. The reported results are obtained by performing 10 repetitions with randomly generated training datasets. All the excluded high-fidelity

points make up the validation set.

Results. Figure 118 shows the variation of the LPSA as the affordable high-fidelity sample budget increases for both the borehole and wing weight functions. A cursory glance is sufficient to clearly notice the benefit offered by the multifidelity MO-GP model over the MO-GP model in terms of the ability to recover the true AS of the high-fidelity function. Across the gamut of ratios of high- to low-fidelity samples, note that the multifidelity model provides a subspace of better quality than the single fidelity model. However, when only a small number of both high- and low-fidelity samples are available ($n = 0.5d,\ 0.75d$ and a ratio of 2), the single-fidelity model is competitive with the multifidelity model for both the functions. This behavior is a result of the insufficiency of training data for both kinds of models as indicated by the large LPSA values. For all values of n, when the difference between the computational costs of the high- and low-fidelity analysis increase, i.e. as the ratio increases, the additional low-fidelity samples start aiding in the recovery of the true AS dramatically. However, when the number of high-fidelity samples starts taking relatively large values, the benefit provided by multifidelity models progressively decreases across all ratios. Lastly, the small interquartile ranges of the measured LPSA over all the training repetitions are an indicator of the consistent behavior of the manifold optimization procedure.

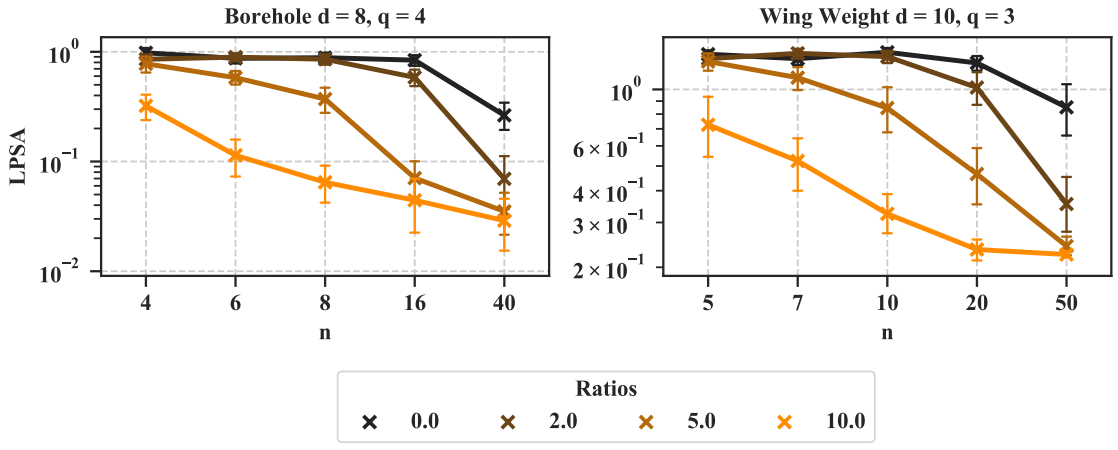


Figure 118: Improvements in Largest Principal Subspace Angle Due to Multifidelity MO-GP

Figure 119 shows the benefit of using multifidelity models in terms of predictive accuracy as measured by the V-NRMSE. The results largely go hand-in-hand with those for the LPSA. Succinctly put, using the multifidelity MO-GP is more likely to be beneficial in the low high-fidelity data regime for analyses that are characterized by a large difference in the computational costs of the high- and low-fidelity functions. The increase in predictive accuracy due to the multifidelity model diminishes as the affordable budget increases and the number of high-fidelity samples start becoming sufficient for a good single-fidelity MO-GP model.

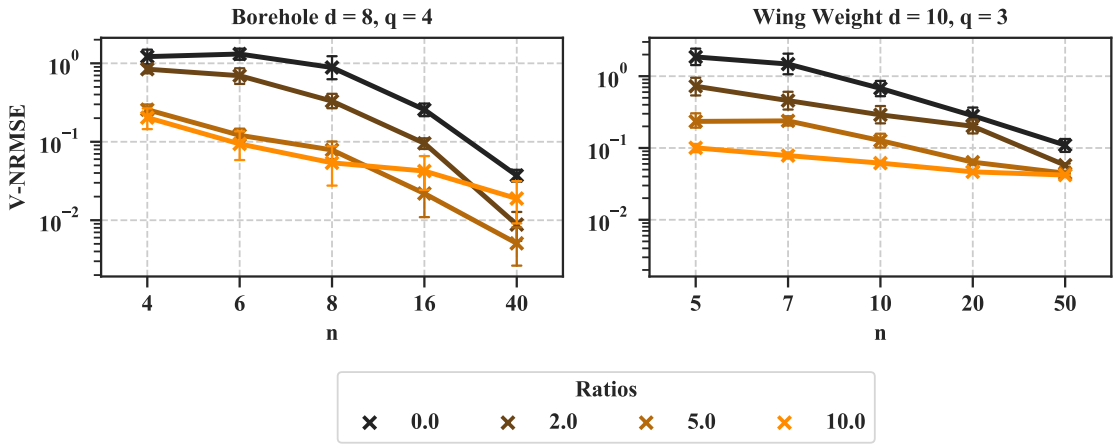


Figure 119: Improvements in V-RMSE Due to Multifidelity MO-GP

6.2.3 Performance of MA-ROM Constructed using Multifidelity MO-GP

Preliminary tests on scalar functions have shown evidence that the multifidelity MO-GP model works as expected. Through the following question, we test its applicability with the MA-ROM for the emulation of field outputs:

Research Question 3.2

When additional low-fidelity data is available, does the proposed model leveraging the manifold alignment-based dimension reduction along with the multifidelity manifold optimization-based Gaussian process regression result in an increase in the predictive accuracy relative to the single fidelity POD and interpolation-based model?

Despite seeing positive results for scalars, we must proceed with caution when considering the multifidelity MO-GP's application with the MA-ROM. Note that in the previous section, we observed that only in the low-data high-fidelity limit does the multifidelity MO-GP provide a relatively significant advantage over the MO-GP model. This would imply that the number of high-fidelity samples must be extremely small to avail the benefits of using the multifidelity MO-GP in the shared latent space obtained by manifold alignment. In such a case, it would be likely that the small number of high-fidelity snapshots is insufficient to capture the physical features of the underlying analysis, i.e. would result in an unacceptably large projection error on the validation set. The minimum sample size requirement is in fact decided by the complexity of the high-fidelity model when dealing with the construction of ROMs. From the observations for applications on practical problems (see chapter 5) so far, we have seen that for analyses with high-dimensional inputs, the smallest number of high-fidelity snapshots for a good predictive ROM is amply sufficient for the MO-GP model. With these thoughts in mind, we formulate the following hypothesis:

Hypothesis 3.2: *For the emulation of field outputs using POD and interpolation-based ROMs, the sample size requirement for a good POD basis, i.e. one that spans the solutions space, is typically sufficient for discovery of the low dimensional input subspace accurately using the MO-GP regression model. If the size of the high-fidelity training set is either just sufficient or barely smaller than what is required for spanning the solution space and the low-dimensional representations of the high- and low-fidelity data have similar low-dimensional input subspaces, additional low-fidelity samples will either provide marginal or no benefit when the number of design variables is large.*

To confirm our expectation, MA-ROMs are constructed to emulate the pressure coefficient field for the 15 variable subsonic airfoil problem. As discussed above, it is not worthwhile to test the proposed methodology for high-fidelity sample sizes that are either too small or too large. Therefore, we construct datasets such that the number of high-fidelity samples is just shy of the minimum size that was found to be sufficient for a good

single-fidelity ROM (as seen in chapter 5). To assess the benefit (or detriment) of low-fidelity data when sufficient high-fidelity samples are available, we consider a case where enough high-fidelity samples are available for the single-fidelity ROM.

Since we do not have access to the gradients for the practical airfoil problem, the performance assessment relies exclusively on the field-level predictive accuracy. All the models are trained 10 times using randomly generated training datasets. We use the MFE (eq. (113)) and the MRE (eq. (115)) computed on the training and validation sets to measure the predictive accuracy of the models.

Details of the Experiment. This experiment begins with a large database of high- and low-fidelity field output data evaluated on a common DoE. For this experiment, we consider two levels for the number of high-fidelity samples. The first case represents a scenario where the budget for the high-fidelity analysis affords training datasets that have a size just below the required number of samples for a good single fidelity ROM. The second case represents a situation where the number of high-fidelity samples is just sufficient for a good single-fidelity ROM.

It is assumed that the dimensionality of the shared AS for each coefficient in the aligned latent space is identical to that of the high-fidelity output's single fidelity POD coefficients. First, 10 repetitions of training datasets containing an appropriate number of high- and low-fidelity samples are generated for each sample size combination setting. Then, for each training dataset depending on whether the training set is single or multifidelity, either the POD and interpolation-based ROM method or the MA-ROM method is invoked to train a ROM. The error metrics are computed on the respective validation sets. Note that when both high- and low-fidelity samples are present in a training set, the low-fidelity data contains solutions evaluated at the design locations contained in the high-fidelity dataset. The additional low-fidelity samples are randomly chosen. This is to ensure compatibility with the requirements of the MA-ROM.

Results. Recall that the 15 variable airfoil problem requires a rank-13 truncation to achieve an RIC of 99.9%. Figure 120 shows the evolution of the field level error metrics

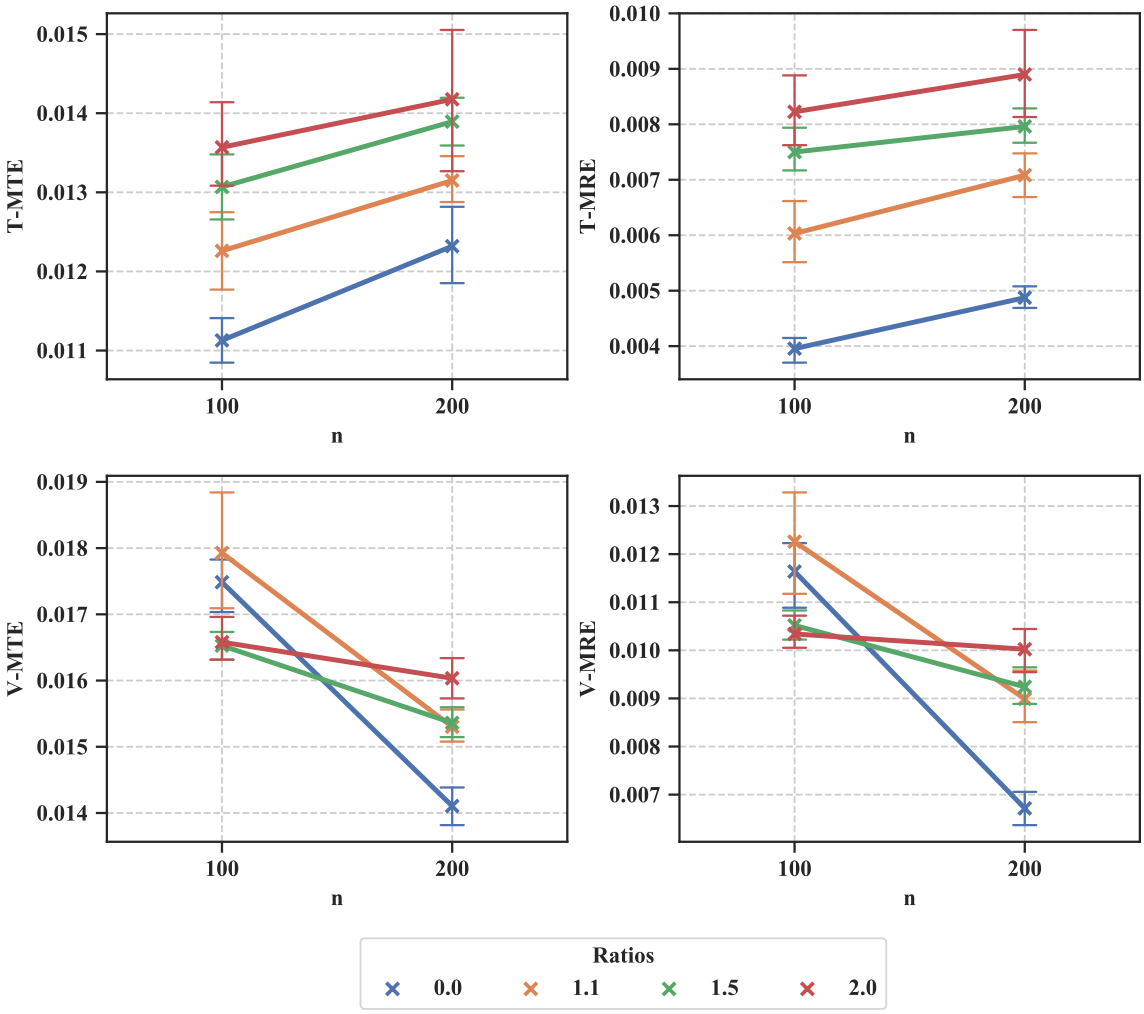


Figure 120: Predictive Performance of the MA-ROM Constructed using the Multifidelity MO-GP

as the affordable budget for high-fidelity samples increases. In-line with our hypothesis, we observe that the error metrics on the validation set are smaller for the MA-ROM when $n = 100$ and $1.5n$ to $2n$ low-fidelity samples are additionally available. However, as soon as $n = 200$, i.e. the number of high-fidelity samples is sufficient for a good POD and interpolation-based ROM using the MO-GP, observe that the presence of additional low-fidelity samples appears to be detrimental for the MA-ROM. In fact, larger number of low-fidelity samples cause the MA-ROM models to perform worse. The reason for this behavior can perhaps be explained by the fact that the AS for the aligned low-fidelity latent space coordinates and high-fidelity latent space coordinates are significantly different. Any

additional low-fidelity samples beyond a sufficient number of high-fidelity samples required for constructing a single-fidelity ROM only causes the multifidelity MO-GP to find an input subspace that is worse in quality than the one comfortably found by the MO-GP model.

6.3 Summary and Concluding Remarks

This chapter proposed a multifidelity extension to the MO-GP model and assessed the feasibility of its ability to refine the estimate of the low-dimensional subspace in which a high-dimensional function varies when the available training data is insufficient. It achieved its goal by making use of a relatively large set of low-fidelity data in addition to sparse high-fidelity data. The model is essentially a multifidelity GP on a Grassmann manifold. The coordinates in the low-dimensional subspace (parametrized by the Grassmann manifold) are obtained by projecting the inputs of both the low- and high-fidelity data using a common projection matrix (a point on the Grassmann manifold). The training procedure simultaneously obtains an MLE of both the multifidelity GP's hyperparameters and the common projection matrix by maximizing the log-marginal likelihood of observing the high- and low-fidelity data in the low-dimensional subspace. Identical to the MO-GP's formulation, the unique formulation is naturally posed as an optimization problem on a product manifold.

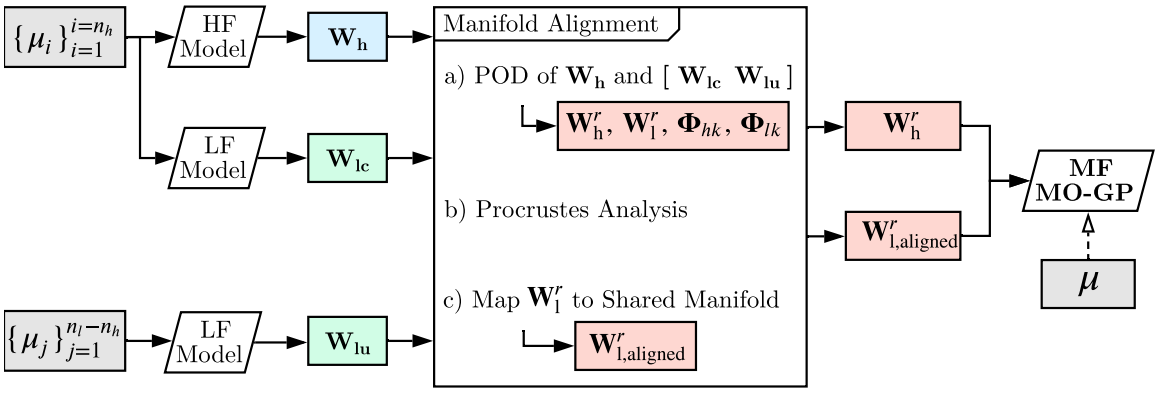


Figure 121: The MA-ROM with the Multifidelity MO-GP

An initial and by no means exhaustive test of its application with the MA-ROM was found to be marginally effective when the number of high-fidelity field samples is smaller

than the minimum required for a good high-fidelity basis set. However, on introducing low-fidelity samples when sufficient high-fidelity data is available, the models perform worse. This is mostly due to a significant difference in the active subspaces of the aligned latent space coordinates of the low-fidelity outputs and the latent space coordinates of the high-fidelity outputs. Note that the hypothesis did not expect the predictive error to worsen. But it assumes that the active subspaces of the aligned high- and low-fidelity latent spaces are similar. In problems this work is concerned with, where the gradients for the latent space coordinates are unavailable, one has to unfortunately rely on the knowledge of the problem to reasonably assume whether the low-dimensional input subspaces for the analyses of differing fidelities are similar. An initial empirical assessment of the methods under this research area has shown promise as hypothesized but warrants a thorough investigation. Figure 121 shows a flowchart depicting the procedure for training the MA-ROM using the multifidelity MO-GP regression model. The hypothesis is re-stated here for convenience.

Hypothesis 3: *The combination of a multifidelity dimension reduction method that finds a common low-dimensional representation of the field outputs and a multifidelity regression model that aids in the accurate discovery of a low-dimensional structure in the inputs will improve the predictive accuracy of ROMs when limited high-fidelity data is available.*

CHAPTER VII

CONCLUSION

The journey and experience of developing the thesis and writing this dissertation has been personally fulfilling and joyful. The content in the dissertation touches upon slivers of concepts in the vast fields of numerical linear algebra, machine learning, and approximation theory. If one were to state it in a single sentence, the main goal of the effort was to highlight the advantages of the cross-fertilization of topics in applied mathematics and aerospace engineering through demonstrations on relevant practical large-scale problems faced by engineers tasked with designing complex systems. While the methods proposed in this dissertation can by no means qualify as a perfect solution, we sincerely hope that it has been able to provide some insights into pertinent issues and challenges, at the very least.

Chapter Outline. The intent for this chapter is to serve as a high-level, yet detailed technical summary of the work performed in the previous chapters. It begins with a re-iteration of the progression of thought (and the accompanying technical work) for each of the three research areas and then summarizes the key findings. Then, we list the contributions made through the gathered empirical evidence. Finally, we critically examine the limitations of all the proposed methods and suggest avenues for future work.

This chapter is especially for readers who wish to obtain a quick compiled summary of all the work that been carried out in this dissertation. Each research area's summary also appears as the last section in its respective chapter. We have merely compiled and reproduced it here for convenience. In addition, the language has been simplified wherever necessary. Before proceeding, please note that figure 122 provides a chart showing all the work presented in the form of question-hypothesis pairs in the respective chapters and figure 123 shows a summary for deciding how to use the methods developed in this work.

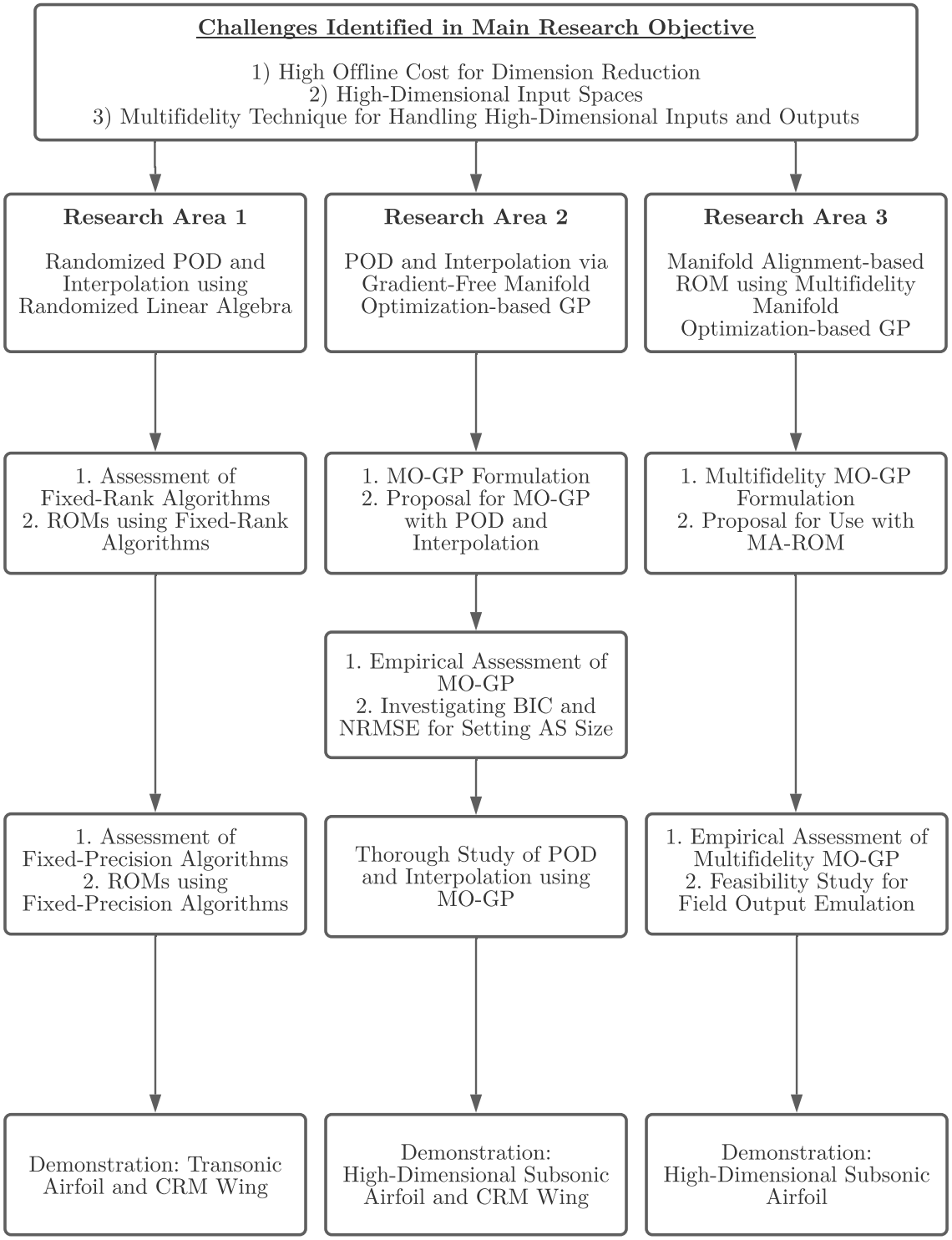


Figure 122: Summary of Research Work

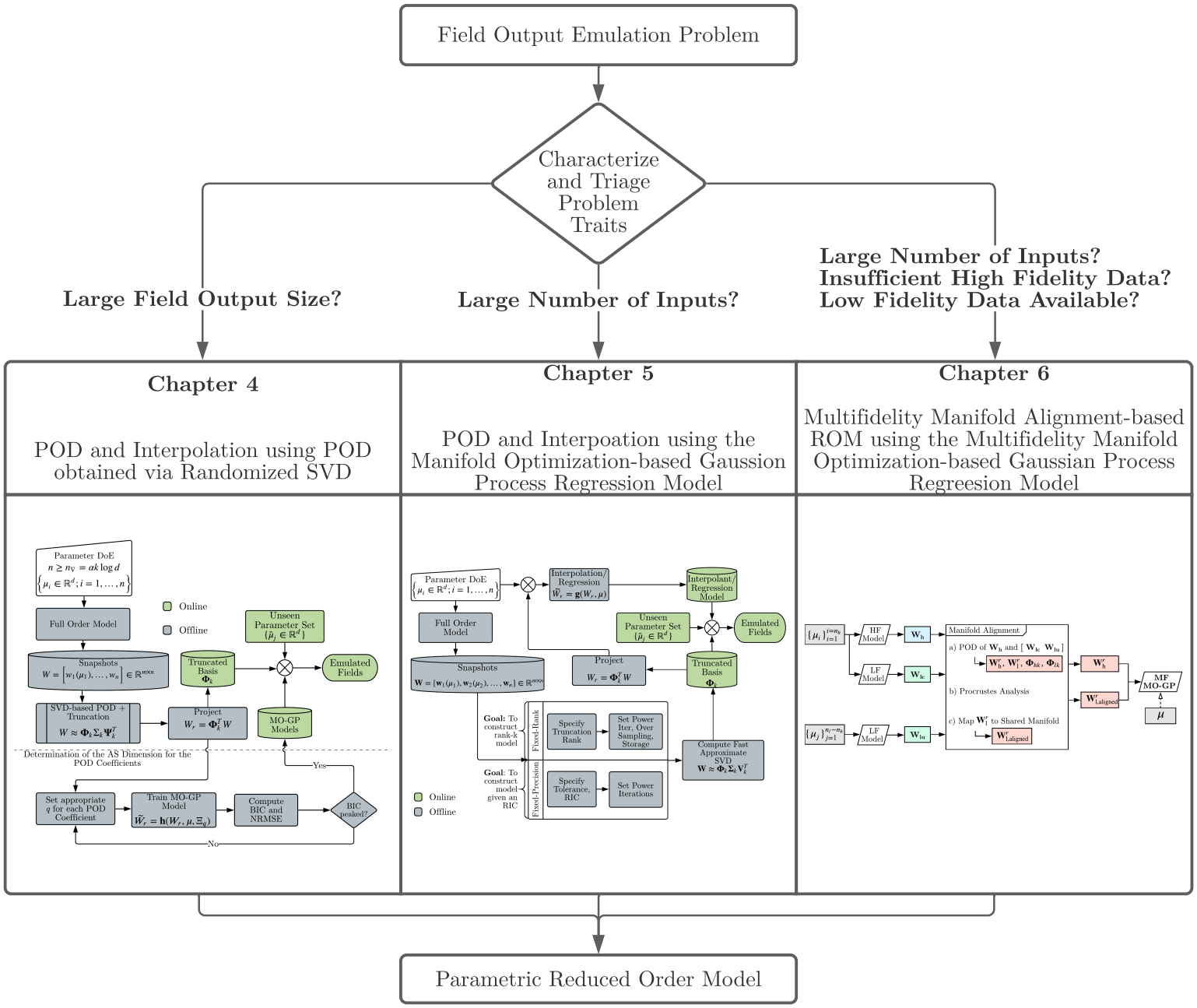


Figure 123: Process Chart

7.1 Detailed Summary of Findings and Limitations

Numerical simulations of systems governed by PDEs are finding use in exploratory design studies due to the accuracy and realism they offer. In part, their use can be viewed as a means to achieve aggressive performance goals by capturing accurate physics. Typical outputs from PDE-based systems are high-dimensional fields. The PDEs themselves are parametric in the design setting i.e. they are dependent on a possibly large number of design/decision variables. In addition, they may also be time-dependent. For successful adoption of PDE-based high-dimensional black-box outputs in *many-query* contexts using approximation methods, we argued that scalar surrogates fail short when dealing with several challenges. Therefore, methods that approximate fields (also called ROMs) must be pursued. All the proposed methods dealt with cases where the system is a black-box in the sense that only outputs and the input parameters that generated them are available.

A thorough literature review revealed three recurring challenges that prevent the adoption of ROMs for truly large and practical problems. This led to the following research objective:

Research Objective

Develop and/or enhance *non-intrusive parametric reduced order modeling methods* to address challenges that limit their viability in many-query contexts involving practical engineering design problems with a focus on analyses that:

1. are expensive to evaluate,
2. output high-dimensional field quantities, and
3. are characterized by a relatively large number of design variables.

The research objective naturally led to the investigation and development of methods in three research areas, each corresponding to one of the three challenge areas. In what follows, we summarize the findings from the work done in each research area.

7.1.1 Research Area 1: High-Offline Cost for Dimension Reduction

In chapter 4, we thoroughly investigated the benefit and effectiveness of randomization as a means to decrease the computational complexity of the dimension reduction step when dealing with large snapshot matrices that may not fit in fast memory. All the experiments and accompanying findings were demonstrated on a canonical problem and two practical flow problems. In any ROM construction exercise, the first step involves performing dimension reduction on a set of pre-computed high-dimensional solutions. The Proper Orthogonal Decomposition (POD) via the Singular Value Decomposition (SVD) is a popular and widely used algorithm for computing the reduced representation (the POD basis set) of the full order model.

Our study began with the use of fixed-rank algorithms where it was found that given a truncation rank, i.e. the amount of compression that must be performed, the randomized SVD (rSVD) and sketching-based rSVD algorithms were both equally effective in computing basis vectors similar to the actual POD basis when run with reasonable values for the algorithm-specific parameters. They did so while significantly reducing the computational complexity of computing the POD. The algorithm-specific parameters provided a means to trade the quality of the POD for gains in computational complexity. Through demonstration on problems with progressively increasing levels of complexity, it was found that large problems with rich physical features generally require relatively conservative values for the algorithm-specific parameters which limit the potential gains in compute time. Nevertheless, the ROMs using the fixed-rank randomized algorithms were competitive in terms of their predictive accuracy to the benchmark POD and interpolation with the deterministic SVD.

Because specification of the truncation rank a priori is a rather harsh requirement, the second part of this chapter investigated a recently proposed fixed-precision randomized SVD (FP-rSVD) algorithm as a candidate to replace the need to require the truncation rank as an input for computing the decomposition. The FP-rSVD algorithm proved effective in finding the approximate numerical rank given an acceptable error tolerance. We argued that while the error tolerance is analogous to the fixed truncation rank, it is a more intuitive parameter to set. In fact, the results showed that the algorithm was effective in finding

the true numerical rank and the actual truncation rank for the canonical problem given a sufficiently small acceptable error tolerance. In general, it was observed that as the problems became progressively challenging and large, the FP-rSVD required a higher number of iterations to find the numerical rank. Moreover, this approximate numerical rank led to inaccurate values of the approximate truncation rank for a given RIC. Fortunately however, for low-rank problems characterized by a small rapid initial drop in the singular value spectrum followed by a relatively slow flat decay, the FP-rSVD algorithm made errors in the less significant trailing singular values and vectors. Consequently, the resulting ROMs are competitive in their predictive accuracy as long as the singular vectors or POD modes associated with the rapidly decaying singular values are accurately captured.

The evidence collected through applications on problems with varying sizes and complexities have empirically shown that techniques from randomized linear algebra can indeed address and efficiently tackle the challenges posed by large datasets to the *offline* cost, thereby validating the hypothesis for this entire research area which is restated below.

Hypothesis 1: *Techniques from Randomized Linear Algebra (RandNLA) will effectively decrease the computational cost associated with the dimension reduction step while incurring a reasonable penalty in terms of accuracy, thereby enabling efficient construction of parametric ROMs for large systems.*

7.1.2 Research Area 2: High-Dimensional Inputs

In chapter 5, we developed and proposed a solution to tackle the challenges posed by large input spaces in the construction of predictive ROMs. Since the interpolation/regression step, which is responsible for encoding parametric dependence, is directly affected by large input spaces, we argued that any advancement must aim at enabling supervised learning in high-dimensions. A thorough literature search put forth mapping-based strategies as the most suitable candidates to address the *curse of dimensionality* in the context of surrogate modeling in high-dimensions. Of specific interest were the methods that circumvented the problems in high-dimensional input spaces by finding a lower dimensional input subspace

and performing the supervised learning in the lower dimensional subspace. The recently proposed Active Subspace Method (ASM) was identified as a method that has enjoyed immense success in practice. However, its reliance on gradients (or Jacobians) rendered it incompatible for our use case of nonintrusive ROMs for which availability of gradients cannot be assumed. In fact, they are typically unavailable and impossible to get without intrusive modification of the black-box. Other competing methods that approximated the gradients using point samples were found to rely on a structured sampling plan in order to provide a guaranteed successful recovery of the AS. Additionally, they required a large number of samples. We argued that any competing method must be able to work with a set of pre-tabulated input-output pairs.

The need for a supervised learning method that does not rely on either the availability of gradients or on a structured sampling plan led to the proposal of a new manifold optimization-based Gaussian process regression model, nicknamed the MO-GP model. The model's formulation redefined the kernel on low-dimensional inputs obtained by projection of the original high-dimensional inputs onto a subspace parametrized using the Grassmann manifold. Both the parameters of the kernel for the GP and the projection subspace were computed simultaneously by maximizing the log-marginal likelihood of observing the training data on a product manifold. The product manifold consisted of a Euclidean part (the GP hyperparameters) and the Grassmann manifold (the projection subspace).

The MO-GP model's performance was analyzed and compared against a competing benchmark model called the GP-AS (nicknamed after how it works) using a large suite of test functions. The GP-AS method first trains a Gaussian process model in the original high-dimensional inputs and then finds the low-dimensional subspace (or the Active Subspace) by sampling gradients from the trained Gaussian process model. Using the gradients to compute the Active Subspace (AS) makes this method similar to the ASM in some sense. In general, the results showed that the MO-GP either outperformed or was at least competitive with the GP-AS method. Surprisingly, it was found that the MO-GP model successfully recovered the true AS without requiring the gradients. However, the MO-GP required specification of the size of the input subspace a priori. This requirement made

its use impractical. Finding an appropriately sized subspace was posed as an exercise in model selection under the realm of machine learning. We asserted and demonstrated that the Bayesian Information Criterion (BIC) and the Normalized Root Mean Squared Error (NRMSE) are effective indicators for finding the size of low-dimensional input subspace that balances predictive accuracy and model complexity. The MO-GP model equipped with the BIC and NRMSE-based method to set the size of the low-dimensional input subspace made our proposal competitive with the ASM.

Following the tests on scalar functions, we proceeded with the use of MO-GP to learn the map between the POD coefficients and high-dimensional inputs by discovering an independent input subspace for each coefficient. Using high-dimensional elliptic PDE problems of different sizes with known gradients obtained intrusively, we demonstrated that the MO-GP could successfully find a low-dimensional input subspace for each POD coefficient. The POD and interpolation-based ROMs using MO-GPs were successfully able to predict the field for unseen inputs in high-dimensions with small errors.

Before concluding, we provide some recommendations for the use of MO-GP to construct POD and interpolation-based ROMs based on our observations. Demonstrations on all the test problems revealed the existence of a small AS for the initial dominant POD modes. For the later and more noisy modes, the dimensionality of the AS increased significantly. In fact, the dimensionality of the AS progressively increased for higher mode indices. This observation highlights a cautionary note regarding the proposed methodology's limitation for certain kinds of problems. For some problem where the snapshot matrix displays a weak singular value spectrum, one can easily imagine a situation where the truncation rank is large enough such that the size of the AS corresponding to the trailing POD modes starts becoming comparable to the original size of the inputs. In other words, the variation of the coefficients for the trailing POD modes cannot be captured in low-dimensional subspaces for such problems. The proposed method will clearly fail to perform in such cases because of the inability of GPs to work in high-dimensions. On the other hand, there may also be problems with a slowly decaying singular value spectrum, variation of whose trailing POD modes can be accurately captured in a low-dimensional subspace using the MO-GP.

The proposed methodology is precisely designed to work effectively in such situations. The evidence gathered through the experiments appear to largely be in agreement with the following hypothesis:

Hypothesis 2: *Parametric ROMs for a relatively large number of inputs can be constructed by modifying and leveraging approaches that identify and construct surrogates in a low-dimensional input subspace with a special focus on addressing the specific challenges posed by vector-valued field outputs.*

7.1.3 Research Area 3: Feasibility of a Multifidelity Technique for Handling High-Dimensional Inputs and Outputs

Chapter 6 proposed a multifidelity extension to the MO-GP model and assessed the feasibility of its ability to refine the estimate of the low-dimensional subspace in which a high-dimensional function varies when the available training data is insufficient. It achieved its goal by making use of a relatively large set of low-fidelity data in addition to sparse high-fidelity data. The model is essentially a multifidelity GP on a Grassmann manifold. The coordinates in the low-dimensional subspace (parametrized by the Grassmann manifold) are obtained by projecting the inputs of both the low- and high-fidelity data using a common projection matrix (a point on the Grassmann manifold). The training procedure simultaneously obtains an MLE of both the multifidelity GP's hyperparameters and the common projection matrix by maximizing the log-marginal likelihood of observing the high- and low-fidelity data in the low-dimensional subspace. Identical to the MO-GP's formulation, the unique formulation is naturally posed as an optimization problem on a product manifold.

Thorough tests on two multifidelity scalar functions showed that the multifidelity MO-GP model successfully provided a refined estimate of the low-dimensional input subspace along with a good predictive accuracy for the high-fidelity function with an extremely small number of high-fidelity samples and a large number of inexpensive low-fidelity samples. An initial and by no means exhaustive test of its application with a recently proposed manifold

alignment based multifidelity ROM method was found to be marginally effective when the high-fidelity field samples are smaller in number than the minimum required for a good high-fidelity basis set. However, on introducing low-fidelity samples when sufficient high-fidelity data is available, the models performed worse than the single-fidelity ROM. This is mostly due to a significant difference in the active subspaces for the low-dimensional representations of the low-fidelity outputs and the high-fidelity outputs. While our rather nascent and initial feasibility study shows promise, a thorough assessment is required to unequivocally and confidently validate the hypothesis stated below:

Hypothesis 3: *The combination of a multifidelity dimension reduction method that finds a common low-dimensional representation of the field outputs and a multifidelity regression model that aids in the accurate discovery of a low-dimensional structure in the inputs will improve the predictive accuracy of ROMs when limited high-fidelity data is available.*

7.2 High-Level Summary of Thesis and Contributions

The past several decades have witnessed steady, rapid strides in the ability to perform numerical computations to simulate the behavior of complex engineered systems. Despite the enormous capabilities afforded by computational tools, conceptual engineering design is largely based on algebraic equations that are valid under simplifying assumptions or on scalar surrogate models that predict scalar quantities. The models that result enable cheap and fast calculations revealing important high-level trades, making them highly desirable and convenient. The success of these methods may be attributed to the fact that vehicles grouped together based on their purpose predominantly looked identical, were manufactured using conventional materials, had comparable aerodynamic performance, and employed similar power plants.

As we move towards aggressive goals and unconventional vehicles, the underlying analyses become computationally burdensome, rendering any rapid many-query exercise intractable. Moreover, instead of predicting scalar outputs, the surrogate models must now

predict spatially distributed high-dimensional field outputs to accurately capture the richer underlying physics from high-fidelity governing equations. In addition to enabling rapid design exercises, predicting field outputs using surrogate models finds crucial utility in situations: 1) where proprietary high-fidelity information that vary with inputs must be passed between organizations without dispensing the underlying models, 2) where spatially distributed quantities must be passed between coupled analyses as the design inputs change, 3) where rapid verification and validation must be performed in detailed design phases without relying on physical experimentation while still maintaining the high quality of data provided by physical experiments, and 4) where spatially distributed quantities must be predicted for digital-twinning applications for the monitoring, prognosis, and diagnosis of deployed systems in real-time using incoming operational data as inputs.

However, research on methods for the construction of surrogates has been limited to academic, small-scale problems with inputs and outputs of moderate sizes, i.e. problems that are not representative of practical systems for which these field surrogate models are claimed to be worthwhile. This dissertation therefore, focuses on solving challenges that characterize practical, large-scale problems. The first contribution concerns the ability to compress large datasets with high-dimensional outputs efficiently under a constrained computational budget. The second contribution solves the challenge of tackling large input spaces that characterize most modern day engineered systems. The third contribution tackles the challenge of achieving high predictive accuracy with limited good quality data. The dissertation contains ample one-of-a-kind demonstrations where models that predict field outputs are built on truly practical, large-scale problems. It is hoped that the techniques developed in this thesis faithfully replace physical experiment- or expensive simulation-based time-consuming steps in the design pipelines of modern day complex engineered systems.

The contributions and corresponding archival publications are listed below:

1. **Randomized POD-based Interpolation Method**

   The methodology resulting from the experiments investigating the benefits of using Randomized Linear Algebra for the dimension reduction step constitute the first contribution. In the papers, we present a short summary of the studies performed under

the first research area. Our studies provide insights into the behavior of randomized algorithms when used to construct ROMs. The results can be used by practitioners to inform the choice of algorithm specific parameters for their own applications. The demonstrations on practical problems can be used as benchmarks to set expectations in terms of computational savings and approximation quality for similar large-scale applications.

Publication: Nonintrusive Parametric Reduced Order Models using Randomized Algorithms, SciTech 2020 (***Published***) [206] → AIAA Journal 2020 (***Published***) [204]

2. **POD-based Interpolation Method for High-dimensional Parameter Spaces using Active Subspaces**

   The methodology resulting from the second research area constitutes the second contribution. In short, the methodology proposes the use of the MO-GP model in conjunction with the POD and interpolation-based ROM method to discover low-dimensional input subspaces for each POD coefficient. Our method is unique in that it does not require access to gradient-information, it works with a pre-evaluated set of input-output pairs, and is scalable to problems with large input spaces (relative to the contemporary applications of parametric ROMs).

   Publication: Non-Intrusive Parametric Reduced Order Models with High-Dimensional Inputs via Gradient-Free Active Subspace, SciTech 2020 (***Published***) [203] → AIAA Journal (***In Preparation***).

3. **Manifold Alignment-based Multifidelity for High-dimensional Parameter Spaces using Gradient-Free Active Subspaces**

   As part of the third research area, we proposed a novel multifidelity extension to the MO-GP model to discover a common input subspace given sparse high-fidelity and abundant low-fidelity data. Additionally, we proposed and demonstrated its use with the recently proposed manifold alignment-based ROM. Since, the primary goal of this research area was merely an initial feasibility assessment, no plans currently exist for converting this work into a publication. Archival papers are deferred to the future

pending a detailed investigation which is currently out of scope.

7.3 Recommendations for Future Work

The recommendations for future work stem from either the limitations of the proposed approaches or from the need to test the applicability of the methods on a variety of practical problems. By no means exhaustive, listed below are some potential avenues for future work:

1. All the applications that used randomized algorithms for the dimension reduction step were demonstrated on large but steady-state problems. In order to evaluate the potential for large gains in computational efficiency, one can apply the methodology to time-dependent parametric systems where the snapshot matrix is much larger than it is for steady-state systems.

2. The sketching-based rSVD algorithm is technically designed to work in *streaming data* scenarios. The incoming data-stream (say, from a time-dependent, parametric simulation) can technically be compressed on-the-fly without requiring to store it. A streaming implementation of the algorithm and its benefits can be a potentially interesting investigation especially for real-time ROM construction applications.

3. Recall that the gradient-free MO-GP model used the BIC and NRMSE error metric to set the size of the low-dimensional input subspace. A very fruitful and challenging investigation could be looking into techniques for finding the size of the subspace along with the training procedure. If this is possible, it would eliminate the need to train several models to balance accuracy and parsimony.

4. Lastly, this work only tests the initial feasibility of the multifidelity manifold alignment and MO-GP based ROM. While the primary goal was to check whether the proposal works or not, recommendation for its confident use requires a thorough investigation into its inner workings.

APPENDIX A

SUPPLEMENTARY SUPPORTING FIGURES

In chapter 5, we proposed the use of the Bayesian Information Criterion (BIC) to select the size of the low-dimensional input subspace for training the MO-GP model. The results, i.e. variation of the BIC with input subspace dimension q, shown for the elliptic PDE, airfoil, and CRM wing problems were reported for models trained with training dataset sizes that satisfy the assumption that ensure accuracy of the BIC. Its accuracy and reliability hinges on the assumption that the ratio of the number of training samples to the number of trainable model parameters is large. Satisfying this assumption is important because the expression for the BIC is an approximation of the posterior probability of a model given the training data. Therefore, the expression for the BIC can only be trusted under the simplifying conditions its derivation assumed. However, we also ran several experiments with different training dataset sizes (small and large) and plotted the behavior of the BIC with the number of input dimensions q. This appendix reports the results for all the additional experiments that reinforce the findings but added marginal value in the main body of the dissertation. Figures 124-128 show that the BIC is largely insensitive to the number of training samples because even with seemingly smaller training dataset sizes, the trends across repetitions (indicated by lines of different colors) remain unchanged, i.e. consistently reveal the same information regarding the size of the input subspace.

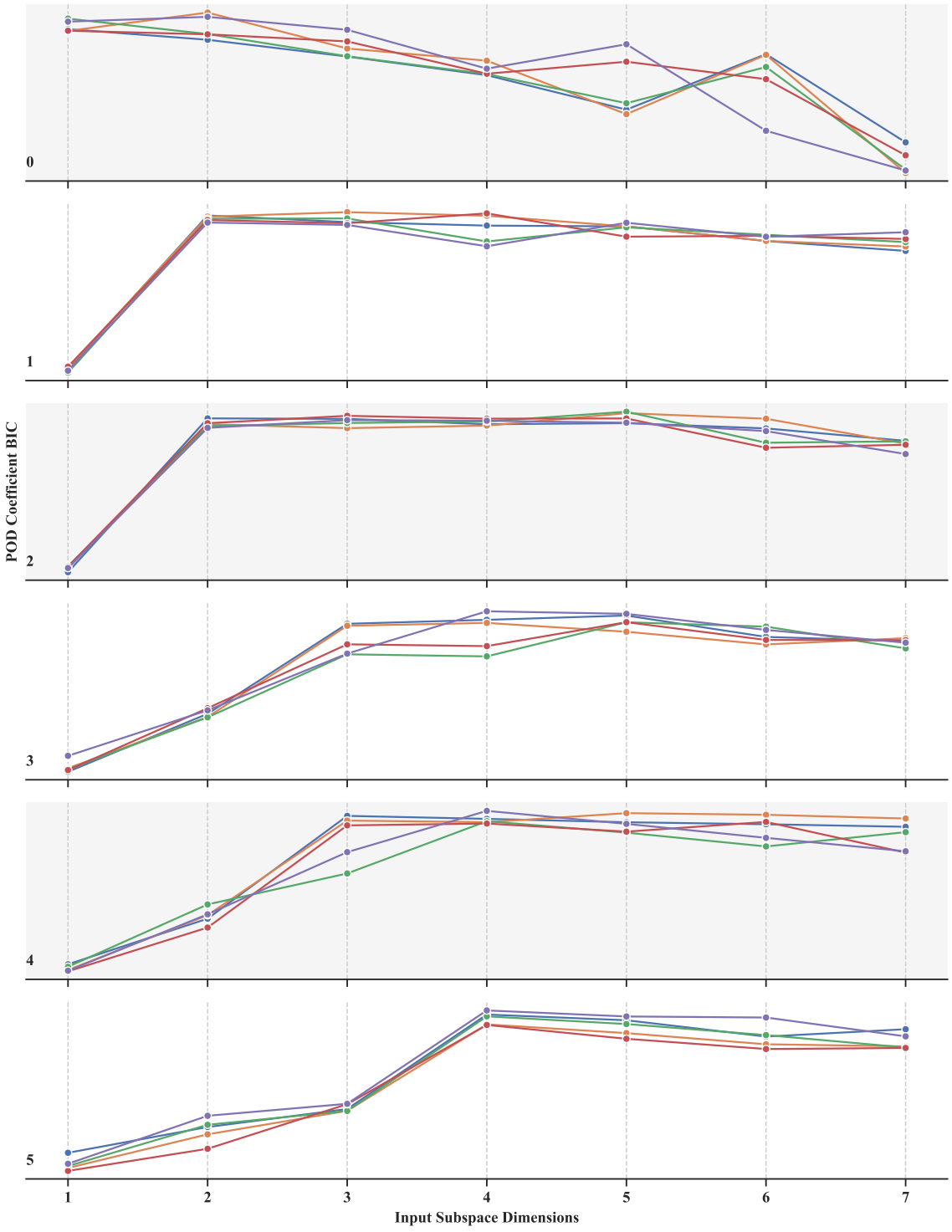


Figure 124: Variation of BIC for the 25 Variable Elliptic PDE Problem with 450 Training Samples. Colors Represent Different Repetitions.

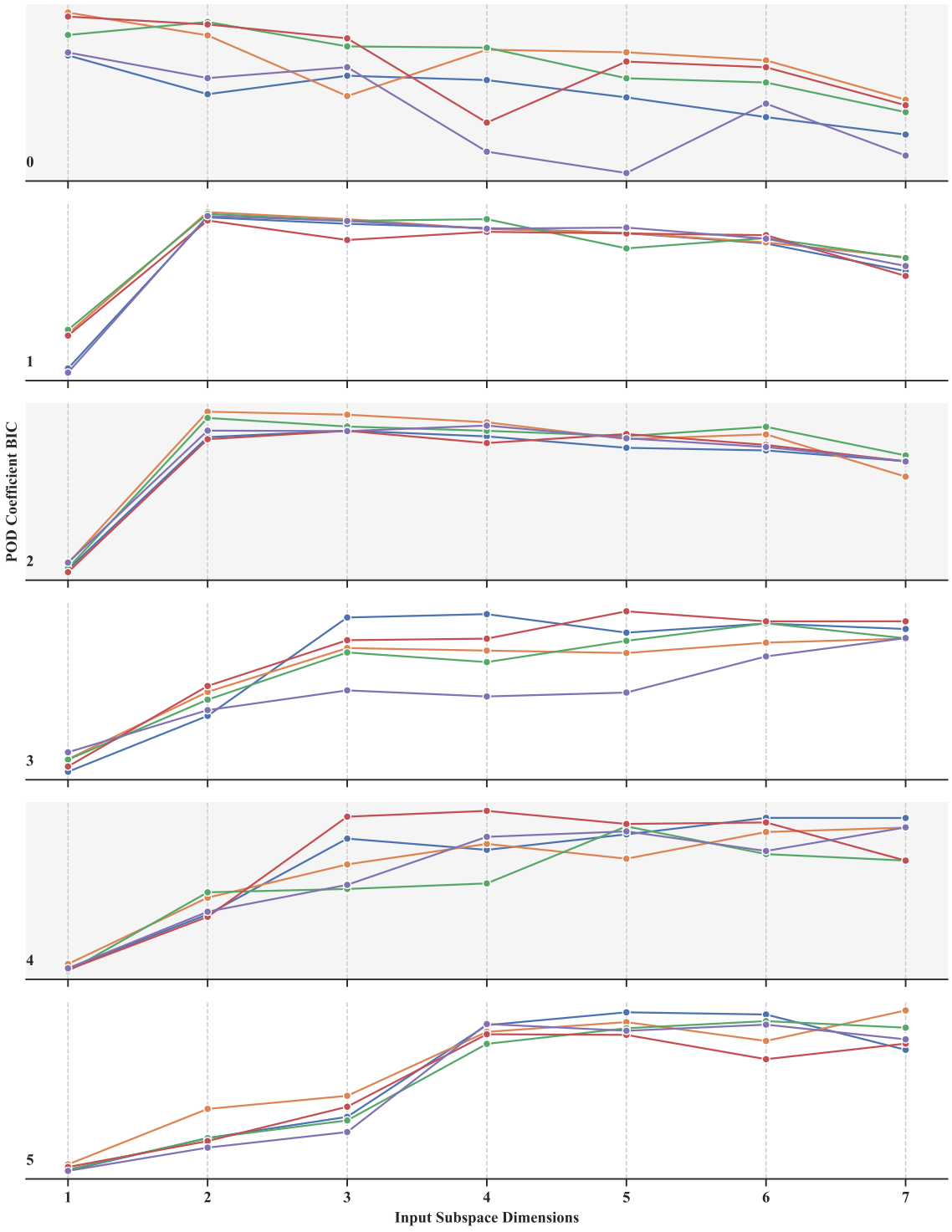


Figure 125: Variation of BIC for the 50 Variable Elliptic PDE Problem with 450 Training Samples. Colors Represent Different Repetitions.

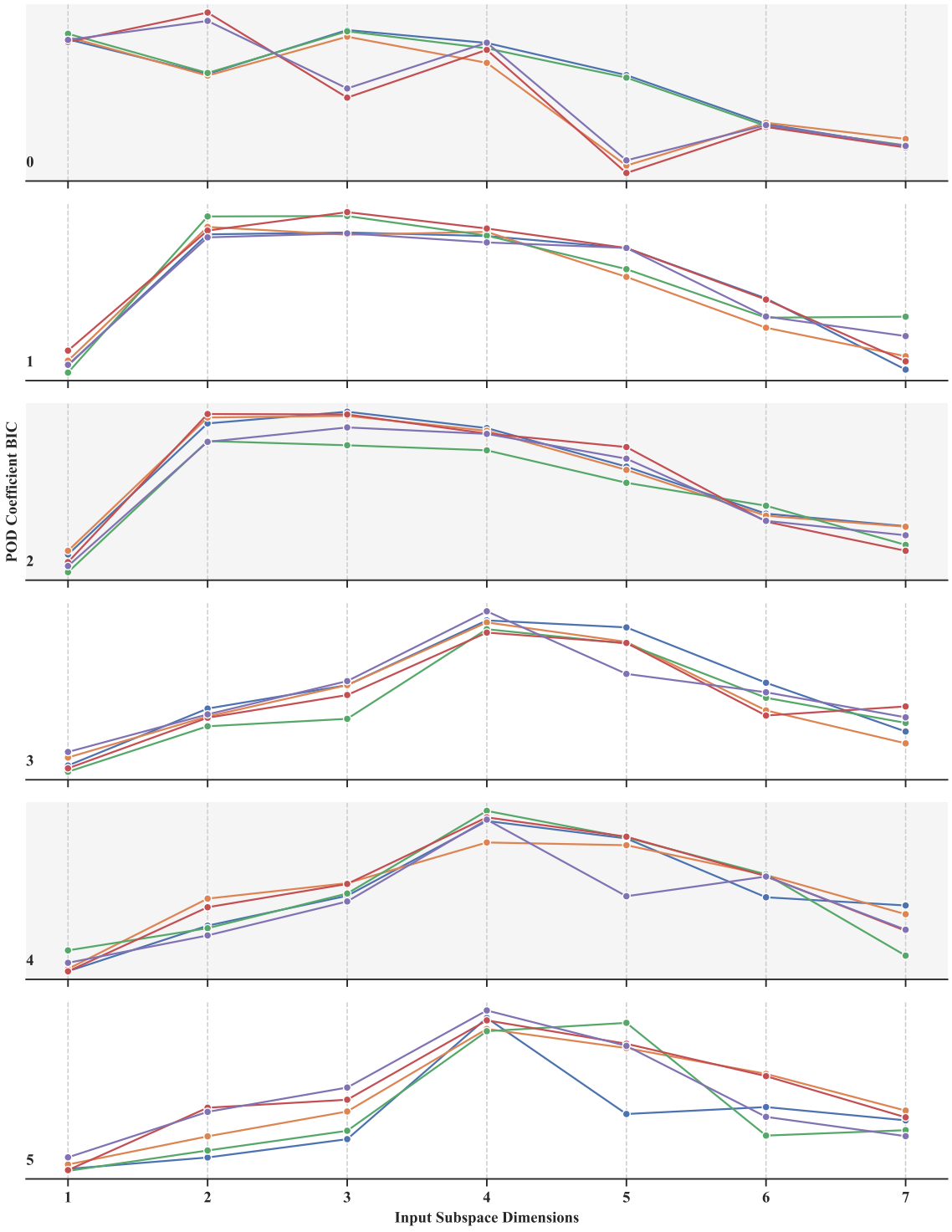


Figure 126: Variation of BIC for the 100 Variable Elliptic PDE Problem with 450 Training Samples. Colors Represent Different Repetitions.

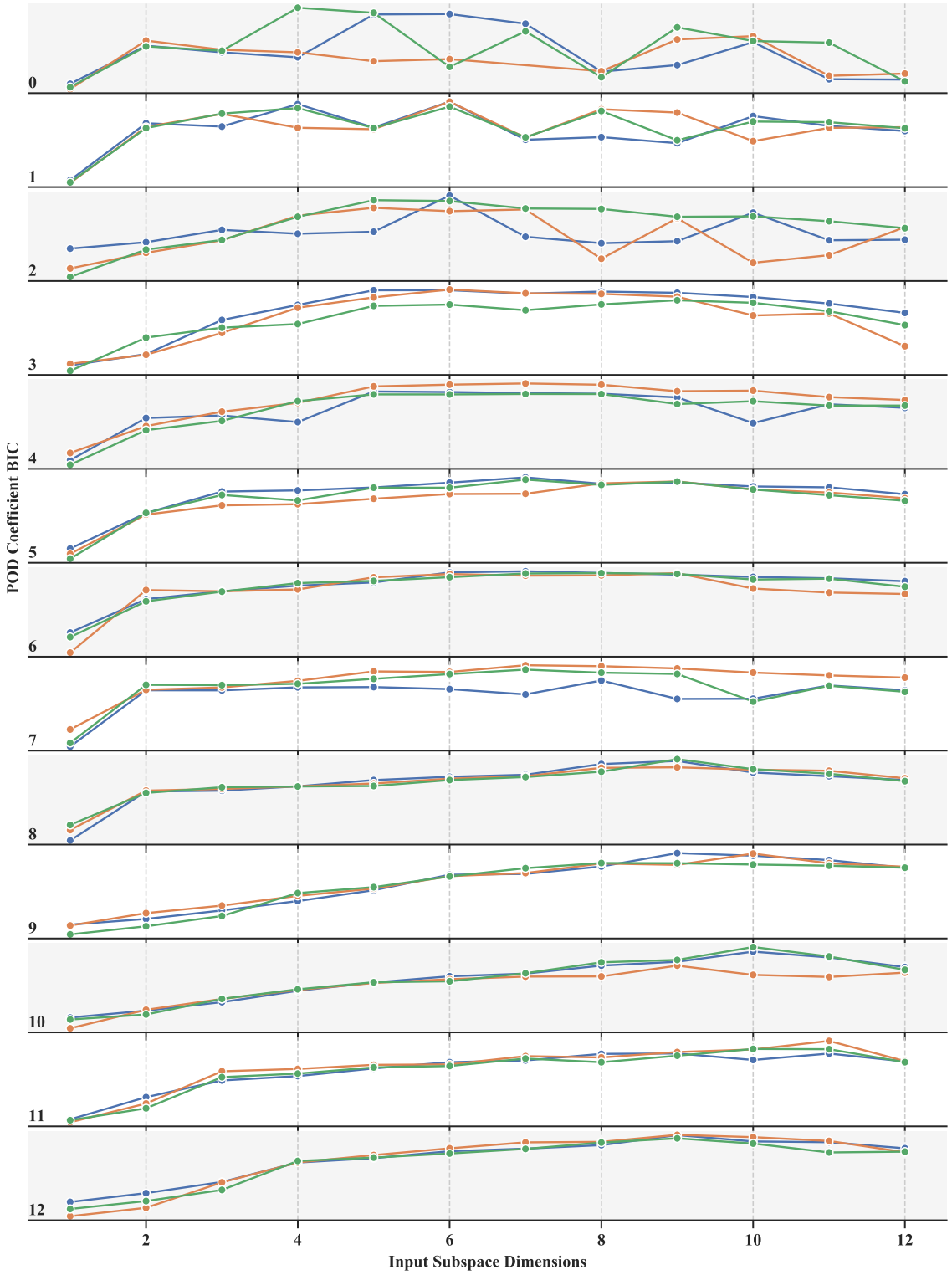


Figure 127: Variation of BIC for the 15 Variable Airfoil Problem with 300 Training Samples. Colors Represent Different Repetitions.

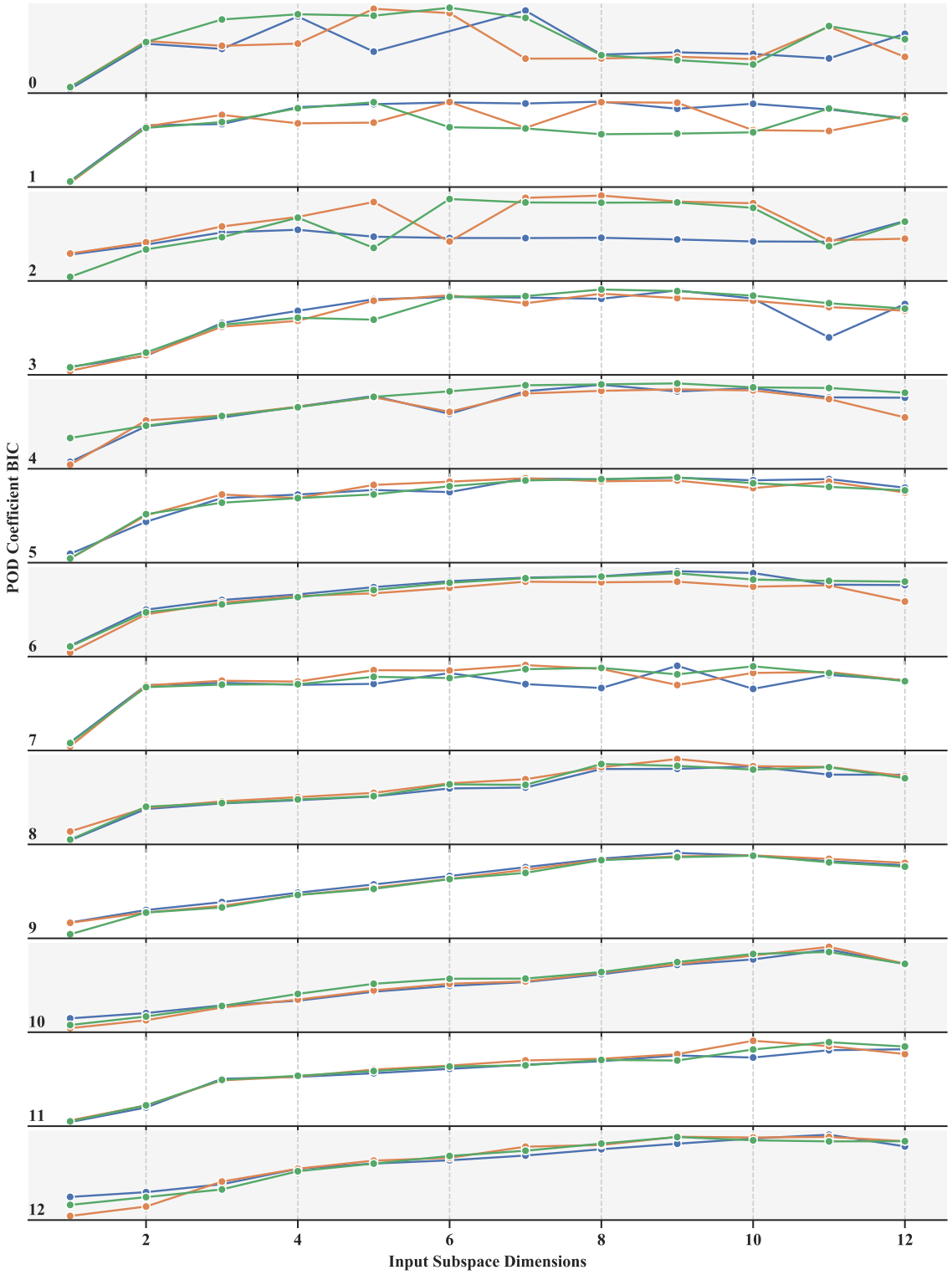


Figure 128: Variation of BIC for the 15 Variable Airfoil Problem with 500 Training Samples. Colors Represent Different Repetitions.

REFERENCES

[1] "Aerodynamic design optimization discussion group." `https://sites.google.com/view/mcgill-computational-aerogroup/adodg`, Accessed: 2020-03-18.

[2] "Aerodynamic design optimization workshop." `http://mdolab.engin.umich.edu/content/aerodynamic-design-optimization-workshop`, Accessed: 2020-03-18.

[3] Absil, P.-A., Mahony, R., and Sepulchre, R., *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[4] Akaike, H., "Information Theory and an Extension of the Maximum Likelihood Principle," pp. 199–213, Springer, New York, NY, 1998.

[5] Albright, B., "Supersonic Simulation."

[6] Alexandrov, N. M., Dennis, J. E., Lewis, R. M., and Torczon, V., "A trust-region framework for managing the use of approximation models in optimization," *Structural Optimization*, vol. 15, pp. 16–23, feb 1998.

[7] Alfred Tze-Mun Leung and Khazaka, R., "Parametric Model Order Reduction Technique For Design Optimization," in *2005 IEEE International Symposium on Circuits and Systems*, no. 1, pp. 1290–1293, IEEE, 2005.

[8] Algazi, V. and Sakrison, D., "On the optimality of the Karhunen-Loève expansion (Corresp.)," *IEEE Transactions on Information Theory*, vol. 15, pp. 319–321, mar 1969.

[9] Alla, A. and Kutz, J. N., "Nonlinear Model Order Reduction via Dynamic Mode Decomposition," *SIAM Journal on Scientific Computing*, vol. 39, pp. B778–B796, jan 2017.

[10] Alla, A. and Kutz, J. N., "Randomized model order reduction," *Advances in Computational Mathematics*, jan 2019.

[11] Amsallem, D., Cortial, J., Carlberg, K., and Farhat, C., "A method for interpolating on manifolds structural dynamics reduced-order models," *International Journal for Numerical Methods in Engineering*, vol. 80, pp. 1241–1258, nov 2009.

[12] Amsallem, D., Cortial, J., and Farhat, C., "Towards Real-Time Computational-Fluid-Dynamics-Based Aeroelastic Computations Using a Database of Reduced-Order Information," *AIAA Journal*, vol. 48, no. 9, pp. 2029–2037, 2010.

[13] Amsallem, D. and Farhat, C., "Interpolation Method for Adapting Reduced-Order Models and Application to Aeroelasticity," *AIAA Journal*, vol. 46, pp. 1803–1813, jul 2008.

[14] Amsallem, D. and Farhat, C., "An Online Method for Interpolating Linear Parametric Reduced-Order Models," *SIAM Journal on Scientific Computing*, vol. 33, pp. 2169–2198, jan 2011.

[15] Annoni, J. and Seiler, P., "A method to construct reduced-order parameter-varying models," *International Journal of Robust and Nonlinear Control*, vol. 27, pp. 582–597, mar 2017.

[16] Antoulas, A. C., *Approximation of Large-Scale Dynamical Systems*. Society for Industrial and Applied Mathematics, jan 2005.

[17] Ashcraft, S. W., Padron, A. S., Pascioni, K. A., Stout Jr, G. W., and Huff, D. L., "Review of propulsion technologies for N+ 3 subsonic vehicle concepts," 2011.

[18] Askham, T., Zheng, P., Aravkin, A., and Kutz, J. N., "Robust and scalable methods for the dynamic mode decomposition," vol. 1, dec 2017.

[19] Astrid, P., Weiland, S., Willcox, K., and Backx, T., "Missing Point Estimation in Models Described by Proper Orthogonal Decomposition," *IEEE Transactions on Automatic Control*, vol. 53, pp. 2237–2251, nov 2008.

[20] Audouze, C., De Vuyst, F., and Nair, P. B., "Reduced-order modeling of parameterized PDEs using time-space-parameter principal component analysis," *International Journal for Numerical Methods in Engineering*, vol. 80, pp. 1025–1057, nov 2009.

[21] Audouze, C., De Vuyst, F., and Nair, P. B., "Nonintrusive reduced-order modeling of parametrized time-dependent partial differential equations," *Numerical Methods for Partial Differential Equations*, vol. 29, pp. 1587–1628, sep 2013.

[22] Bach, C., Ceglia, D., Song, L., and Duddeck, F., "Randomized low-rank approximation methods for projection-based model order reduction of large nonlinear dynamical problems," *International Journal for Numerical Methods in Engineering*, vol. 118, pp. 209–241, apr 2019.

[23] Bach, C., Duddeck, F., and Song, L., "Fixedprecision randomized lowrank approximation methods for nonlinear model order reduction of large systems," *International Journal for Numerical Methods in Engineering*, vol. 119, pp. 687–711, aug 2019.

[24] Bamer, F., Amiri, A. K., and Bucher, C., "A new model order reduction strategy adapted to nonlinear problems in earthquake engineering," *Earthquake Engineering & Structural Dynamics*, vol. 46, pp. 537–559, apr 2017.

[25] Bandler, J. W., Cheng, Q. S., Dakroury, S. A., Mohamed, A. S., Bakr, M. H., Madsen, K., and Søndergaard, J., "Space mapping: The state of the art," *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 1 II, pp. 337–361, 2004.

[26] Barton, R. R., "Metamodels for simulation input-output relations," in *Proceedings of the 24th conference on Winter simulation - WSC '92*, (New York, New York, USA), pp. 289–299, ACM Press, 1992.

[27] Benamara, T., Breitkopf, P., Lepot, I., and Sainvitu, C., "Multi-Fidelity Extension To Non-Intrusive Proper Orthogonal Decomposition Based Surrogates," *Proceedings of the VII European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS Congress 2016)*, no. January 2016, pp. 4129–4145, 2016.

[28] Bendarkar, M. V., Rajaram, D., Cai, Y., Briceno, S. I., and Mavris, D. N., "Evaluating optimal paths for aircraft subsystem electrification in early design," in *AIAA Aviation 2019 Forum*, p. 2802, 2019.

[29] Benner, P., Cohen, A., Ohlberger, M., Willcox, K., and Society for Industrial and Applied Mathematics, *Model Reduction and Approximation.* Philadelphia, PA: Society for Industrial and Applied Mathematics, jul 2017.

[30] Benner, P., Feng, L., Li, S., and Zhang, Y., "Reduced-Order Modeling and ROM-Based Optimization of Batch Chromatography," vol. 103 of *Lecture Notes in Computational Science and Engineering*, pp. 427–435, Cham: Springer International Publishing, 2015.

[31] Benner, P., Gugercin, S., and Willcox, K., "A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems," *SIAM Review*, vol. 57, pp. 483–531, jan 2015.

[32] Benner, P. and Saak, J., "Numerical solution of large and sparse continuous time algebraic matrix Riccati and Lyapunov equations: a state of the art survey," *GAMM-Mitteilungen*, vol. 36, pp. 32–52, aug 2013.

[33] Benner, P., Sachs, E., and Volkwein, S., "Model order reduction for PDE constrained optimization," in *Trends in PDE constrained optimization*, pp. 303–326, Springer, 2014.

[34] Berger, E., Sastuba, M., Vogt, D., Jung, B., and Ben Amor, H., "Estimation of perturbations in robotic behavior using dynamic mode decomposition," *Advanced Robotics*, vol. 29, pp. 331–343, mar 2015.

[35] Berguin, S. H. and Mavris, D. N., "Dimensionality Reduction Using Principal Component Analysis Applied to the Gradient," *AIAA Journal*, vol. 53, no. 4, pp. 1078–1090, 2014.

[36] Berguin, S. H., Rancourt, D., and Mavris, D. N., "Method to Facilitate High-Dimensional Design Space Exploration Using Computationally Expensive Analyses," *AIAA Journal*, vol. 53, no. 12, pp. 3752–3765, 2015.

[37] Bertram, A., Othmer, C., and Zimmermann, R., "Towards Real-time Vehicle Aerodynamic Design via Multi-fidelity Data-driven Reduced Order Modeling," in *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, no. January, (Reston, Virginia), American Institute of Aeronautics and Astronautics, jan 2018.

[38] Bhosekar, A. and Ierapetritou, M., "Advances in surrogate based modeling, feasibility analysis, and optimization: A review," *Computers & Chemical Engineering*, vol. 108, pp. 250–267, 2018.

[39] Bishop, C. M., *Pattern recognition and machine learning*. Springer, 2006.

[40] Bistrian, D. A. and Navon, I. M., "An improved algorithm for the shallow water equations model reduction: Dynamic Mode Decomposition vs POD," *International Journal for Numerical Methods in Fluids*, vol. 78, pp. 552–580, jul 2015.

[41] Bistrian, D. A. and Navon, I. M., "Efficiency of randomised dynamic mode decomposition for reduced order modelling," *International Journal of Computational Fluid Dynamics*, vol. 32, pp. 88–103, mar 2018.

[42] Bistrian, D. A. and Navon, I. M., "Randomized dynamic mode decomposition for nonintrusive reduced order modelling," *International Journal for Numerical Methods in Engineering*, vol. 112, pp. 3–25, oct 2017.

[43] Boyd, S. P. and Vandenberghe, L., *Introduction to applied linear algebra : vectors, matrices, and least squares*.

[44] Brooks, C. L., Karplus, M., and Pettitt, B., *Proteins: a Theoretical Perspective of Dynamics, Structure and Thermodynamics*. New York: Wiley, 1988.

[45] Brunton, B. W., Johnson, L. A., Ojemann, J. G., and Kutz, J. N., "Extracting spatialtemporal coherent patterns in large-scale neural recordings using dynamic mode decomposition," *Journal of Neuroscience Methods*, vol. 258, pp. 1–15, jan 2016.

[46] Brunton, S. L., Brunton, B. W., Proctor, J. L., and Kutz, J. N., "Koopman Invariant Subspaces and Finite Linear Representations of Nonlinear Dynamical Systems for Control," *PLOS ONE*, vol. 11, p. e0150171, feb 2016.

[47] Budišić, M., Mohr, R., and Mezić, I., "Applied Koopmanism," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 22, p. 047510, dec 2012.

[48] Buhmann, M. D., *Radial basis functions: theory and implementations*, vol. 12. Cambridge university press, 2003.

[49] Buhr, A. and Smetana, K., "Randomized Local Model Order Reduction," *SIAM Journal on Scientific Computing*, vol. 40, pp. A2120–A2151, jan 2018.

[50] Bui-Thanh, T., Damodaran, M., and Willcox, K., "Proper Orthogonal Decomposition Extensions for Parametric Applications in Compressible Aerodynamics," in *21st AIAA Applied Aerodynamics Conference*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, jun 2003.

[51] Bui-Thanh, T., Willcox, K., and Ghattas, O., "Model Reduction for Large-Scale Systems with High-Dimensional Parametric Input Space," *SIAM Journal on Scientific Computing*, vol. 30, pp. 3270–3288, jan 2008.

[52] Bui-Thanh, T., Willcox, K., and Ghattas, O., "Parametric Reduced-Order Models for Probabilistic Analysis of Unsteady Aerodynamic Applications," *AIAA Journal*, vol. 46, pp. 2520–2529, oct 2008.

[53] Bui-Thanh, T., *Model-constrained optimization methods for reduction of parameterized large-scale systems*. PhD thesis, 2007.

[54] Bui-Thanh, T., Damodaran, M., and Willcox, K. E., "Aerodynamic Data Reconstruction and Inverse Design Using Proper Orthogonal Decomposition," *AIAA Journal*, vol. 42, pp. 1505–1516, aug 2004.

[55] Cai, Y., Rajaram, D., and Mavris, D. N., "Multi-mission multi-objective optimization in commercial aircraft conceptual design," in *AIAA Aviation 2019 Forum*, p. 3577, 2019.

[56] Cao, C., Cai, J., Qu, K., and Li, J., "An Efficient Multistep ROM Method for Prediction of Flows over Airfoils," in *55th AIAA Aerospace Sciences Meeting*, no. January, (Reston, Virginia), pp. 1–14, American Institute of Aeronautics and Astronautics, jan 2017.

[57] Carlberg, K., Amsallem, D., Avery, P., Zahr, M., and Farhat, C., "The GNAT nonlinear model reduction method and its application to fluid dynamics problems," in *6th AIAA Theoretical Fluid Mechanics Conference*, no. June, (Reston, Virigina), American Institute of Aeronautics and Astronautics, jun 2011.

[58] Carlberg, K., Barone, M., and Antil, H., "Galerkin v. least-squares Petrov-Galerkin projection in nonlinear model reduction," *Journal of Computational Physics*, vol. 330, pp. 693–734, feb 2017.

[59] Carlberg, K., Bou-Mosleh, C., and Farhat, C., "Efficient non-linear model reduction via a least-squares Petrov-Galerkin projection and compressive tensor approximations," *International Journal for Numerical Methods in Engineering*, vol. 86, pp. 155–181, apr 2011.

[60] Carlberg, K. and Farhat, C., "A Compact Proper Orthogonal Decomposition Basis for Optimization-Oriented Reduced-Order Models," in *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, no. September, (Reston, Virigina), American Institute of Aeronautics and Astronautics, sep 2008.

[61] Carlberg, K. and Farhat, C., "A low-cost, goal-oriented compact proper orthogonal decomposition' basis for model reduction of static systems," *International Journal for Numerical Methods in Engineering*, vol. 86, pp. 381–402, apr 2011.

[62] Carlberg, K. T., *MODEL REDUCTION OF NONLINEAR MECHANICAL SYSTEMS VIA OPTIMAL PROJECTION AND TENSOR APPROXIMATION*. PhD thesis, 2011.

[63] Castellani, M., Lemmens, Y., and Cooper, J. E., "Parametric reduced-order model approach for simulation and optimization of aeroelastic systems with structural nonlinearities," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 230, pp. 1359–1370, jun 2016.

[64] Chaturantabut, S. and Sorensen, D. C., "Nonlinear Model Reduction via Discrete Empirical Interpolation," *SIAM Journal on Scientific Computing*, vol. 32, pp. 2737–2764, jan 2010.

[65] Chen, W., Hesthaven, J. S., Junqiang, B., Qiu, Y., Yang, Z., and Tihao, Y., "Greedy Nonintrusive Reduced Order Model for Fluid Dynamics," *AIAA Journal*, vol. 56, pp. 4927–4943, dec 2018.

[66] Chen, W., Junqiang, B., and Hesthaven, J. S., "An iterative approach to improve non-intrusive reduced-order models efficiency for parameterized problems," in *21st AIAA International Space Planes and Hypersonics Technologies Conference, Hypersonics 2017*, American Institute of Aeronautics and Astronautics Inc, AIAA, 2017.

[67] Choi, S., Alonso, J. J., and Kroo, I. M., "Two-Level Multifidelity Design Optimization Studies for Supersonic Jets," *Journal of Aircraft*, vol. 46, no. 3, pp. 776–790, 2009.

[68] Choi, Y., Amsallem, D., and Farhat, C., "Gradient-based Constrained Optimization Using a Database of Linear Reduced-Order Models," pp. 1–28, jun 2015.

[69] Cohen, A., Daubechies, I., DeVore, R., Kerkyacharian, G., and Picard, D., "Capturing Ridge Functions in High Dimensions from Point Queries," *Constructive Approximation*, vol. 35, pp. 225–243, apr 2012.

[70] Coleman, K. D., Lewis, A., Smith, R. C., Williams, B., Morris, M., and Khuwaileh, B., "Gradient-Free Construction of Active Subspaces for Dimension Reduction in Complex Models with Applications to Neutronics," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 7, pp. 117–142, jan 2019.

[71] Collier, F., "Concepts and Technologies for Green Aviation," tech. rep., NASA, 2009.

[72] Constantine, P. G., *Active Subspaces*. Philadelphia, PA: Society for Industrial and Applied Mathematics, mar 2015.

[73] Constantine, P. G., *Active Subspaces - Emerging Ideas for Dimension Reduction in Parameter Studies*, vol. 8. 2017.

[74] Constantine, P. G., Dow, E., and Wang, Q., "Active Subspace Methods in Theory and Practice: Applications to Kriging Surfaces," *SIAM Journal on Scientific Computing*, vol. 36, pp. A1500–A1524, jan 2014.

[75] Constantine, P. G., Eftekhari, A., and Wakin, M. B., "Computing Active Subspaces Efficiently with Gradient Sketching," jun 2015.

[76] Constantine, P. G., Kent, C., and Bui-Thanh, T., "Accelerating Markov chain Monte Carlo with active subspaces," *SIAM Journal on Scientific Computing*, vol. 38, pp. A2779–A2805, sep 2016.

[77] Cook, P., Mcdonald, M., and Firmin, M., "Experimental data base for computer program assessment," *AGARD AR-138*, 1979.

[78] Daberkow, S. M., Daberkow, D. D., and Mavris, D. N., "New Approaches to Conceptual and Preliminary Aircraft Design: A Comparative Assessment of a Neural Network Formulation and a Response Surface Methodology," 1998.

[79] Decker, K., Schwartz, H. D., and Mavris, D., "Dimensionality Reduction Techniques Applied to the Design of Hypersonic Aerial Systems," American Institute of Aeronautics and Astronautics (AIAA), jun 2020.

[80] Degroote, J., Vierendeels, J., and Willcox, K., "Interpolation among reduced-order matrices to obtain parameterized models for design, optimization and probabilistic analysis," *International Journal for Numerical Methods in Fluids*, no. June 2009, pp. n/a–n/a, 2009.

[81] Dolgov, S. and Stoll, M., "Low-Rank Solution to an Optimization Problem Constrained by the Navier–Stokes Equations," *SIAM Journal on Scientific Computing*, vol. 39, pp. A255–A280, jan 2017.

[82] Drmač, Z., Mezić, I., and Mohr, R., "Data Driven Modal Decompositions: Analysis and Enhancements," *SIAM Journal on Scientific Computing*, vol. 40, pp. A2253–A2285, jan 2018.

[83] Eckart, C. and Young, G., "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, pp. 211–218, sep 1936.

[84] Economon, T. D., Palacios, F., Copeland, S. R., Lukaczyk, T. W., and Alonso, J. J., "SU2: An Open-Source Suite for Multiphysics Simulation and Design," *AIAA Journal*, vol. 54, no. 3, pp. 1–19, 2015.

[85] Edelman, A., Arias, T. A., and Smith, S. T., "The Geometry of Algorithms with Orthogonality Constraints," *SIAM Journal on Matrix Analysis and Applications*, vol. 20, pp. 303–353, jan 1998.

[86] Eftekhari, A., B. Wakin, M., Li, P., and G. Constantine, P., "Randomized learning of the second-moment matrix of a smooth function," *Foundations of Data Science*, vol. 1, no. 3, pp. 329–387, 2019.

[87] Eftekhari, A., Wakin, M. B., and Ward, R. A., "MCˆ2: A Two-Phase Algorithm for Leveraged Matrix Completion," sep 2016.

[88] Eldred, M. and Dunlavy, D., "Formulations for Surrogate-Based Optimization with Data Fit, Multifidelity, and Reduced-Order Models," in *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, sep 2006.

[89] Erichson, N. B., Brunton, S. L., and Kutz, J. N., "Compressed Singular Value Decomposition for Image and Video Processing," in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pp. 1880–1888, IEEE, oct 2017.

[90] Erichson, N. B., Mathelin, L., Brunton, S. L., and Kutz, J. N., "Randomized Dynamic Mode Decomposition," feb 2017.

[91] Everson, R. and Sirovich, L., "KarhunenLoeve procedure for gappy data," *JOSA A*, vol. 12, no. 8, pp. 1657–1664, 1995.

[92] Fang, K., Li, R.-z., and Sudjianto, A., *Design and modeling for computer experiments*. Chapman & Hall/CRC, 2006.

[93] Flaig, A., "Airbus A380: Solutions to the Aerodynamic Challenges of Presentation overview," Tech. Rep. January, Airbus, 2008.

[94] Fornasier, M., Schnass, K., and Vybiral, J., "Learning Functions of Few Arbitrary Linear Parameters in High Dimensions," *Foundations of Computational Mathematics*, vol. 12, pp. 229–262, apr 2012.

[95] Forrester, A. I. and Keane, A. J., "Recent advances in surrogate-based optimization," *Progress in Aerospace Sciences*, vol. 45, no. 1-3, pp. 50–79, 2009.

[96] Forrester, A. I., Sóbester, A., and Keane, A. J., "Multi-fidelity optimization via surrogate modelling," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 463, pp. 3251–3269, dec 2007.

[97] Fossati, M., "Evaluation of Aerodynamic Loads via Reduced-Order Methodology," *AIAA Journal*, vol. 53, no. 8, pp. 2389–2405, 2015.

[98] Franz, T., Zimmermann, R., Görtz, S., and Karcher, N., "Interpolation-based reduced-order modelling for steady transonic flows via manifold learning," *International Journal of Computational Fluid Dynamics*, vol. 28, pp. 106–121, mar 2014.

[99] G, S., ""Estimating the Dimension of a Model."," *The Annals of Statistics*, vol. 6, no. 2, pp. 461 – 464, 1978.

[100] Ghodsi, A., "Dimensionality Reduction A Short Tutorial,"

[101] Ghosh, S., Kristensen, J., Zhang, Y., Subber, W., and Wang, L., "A strategy for adaptive sampling of multi-fidelity gaussian processes to reduce predictive uncertainty," in *ASME 2019 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, American Society of Mechanical Engineers Digital Collection, 2019.

[102] Giunta, A. and Eldred, M., "Implementation of a trust region model management strategy in the DAKOTA optimization toolkit," in *8th Symposium on Multidisciplinary Analysis and Optimization*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, sep 2000.

[103] Glende, W. L., "The Boeing 777 : A Look Back," 2000.

[104] Gogu, C., "Improving the efficiency of large scale topology optimization through on-the-fly reduced order model construction," *International Journal for Numerical Methods in Engineering*, vol. 101, pp. 281–304, jan 2015.

[105] Golub, G. and Loan, C. V., *Matrix computations*. 2012.

[106] Goron, G., Duca, R., Sarojini, D., Shah, S., Chakraborty, I., Briceno, S. I., and Mavris, D. N., "A simulation-based framework for structural loads assessment during dynamic maneuvers," in *17th AIAA Aviation Technology, Integration, and Operations Conference*, p. 3767, 2017.

[107] Görtz, S., Ripepi, M., Franz, T., Abu-Zurayk, M., and Zimmermann, R., "Reduced Order Models for Aerodynamic Applications, Loads and MDO in colloboration with," pp. 1–19, 2016.

[108] Grosek, J. and Kutz, J. N., "Dynamic Mode Decomposition for Real-Time Background/Foreground Separation in Video," apr 2014.

[109] Guo, M. and Hesthaven, J. S., "Data-driven reduced order modeling for time-dependent problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 345, pp. 75–99, mar 2019.

[110] Guy, H., Alexanderian, A., and Yu, M., "A distributed active subspace method for scalable surrogate modeling of function valued outputs," aug 2019.

[111] Haasdonk, B., "Chapter 2: Reduced Basis Methods for Parametrized PDEsA Tutorial Introduction for Stationary and Instationary Problems," in *Model Reduction and Approximation*, pp. 65–136, Philadelphia, PA: Society for Industrial and Applied Mathematics, jul 2017.

[112] Halko, N., Martinsson, P. G., and Tropp, J. A., "Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions," *SIAM Review*, vol. 53, pp. 217–288, jan 2011.

[113] Ham, J., Lee, D. D., and Saul, L. K., "Semisupervised alignment of manifolds," *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics*, vol. 10, pp. 120–127, 2005.

[114] Han, Z.-H. and Görtz, S., "Hierarchical Kriging Model for Variable-Fidelity Surrogate Modeling," *AIAA Journal*, vol. 50, no. 9, pp. 1885–1896, 2012.

[115] Hastie, Trevor, Tibshirani, Robert, Friedman, J., *The Elements of Statistical Learning The Elements of Statistical LearningData Mining, Inference, and Prediction, Second Edition.* 2009.

[116] Henneron, T., Montier, L., Pierquin, A., and Clenet, S., "Comparison of DEIM and BPIM to Speed Up a POD-Based Nonlinear Magnetostatic Model," *IEEE Transactions on Magnetics*, vol. 53, pp. 1–4, jun 2017.

[117] Hesthaven, J. and Ubbiali, S., "Non-intrusive reduced order modeling of nonlinear problems using neural networks," *Journal of Computational Physics*, vol. 363, pp. 55–78, jun 2018.

[118] Holmes, P., Lumley, J. L., and Berkooz, G., *Turbulence, Coherent Structures, Dynamical Systems and Symmetry.* Cambridge: Cambridge University Press, 1996.

[119] Hotelling, H., "Analysis of a complex of statistical variables into principal components.," *Journal of Educational Psychology*, vol. 24, no. 6, pp. 417–441, 1933.

[120] Hoyle, D. C., "Automatic PCA dimension selection for high dimensional data and small sample sizes," *Journal of Machine Learning Research*, vol. 9, no. Dec, pp. 2733–2759, 2008.

[121] Ji, W., Wang, J., Zahm, O., Marzouk, Y. M., Yang, B., Ren, Z., and Law, C. K., "Shared low-dimensional subspaces for propagating kinetic uncertainty to multiple outputs," *Combustion and Flame*, 2018.

[122] John D. Anderson, J., "Aircraft Performance and Design Anderson," 1999.

[123] Jovanović, M. R., Schmid, P. J., and Nichols, J. W., "Sparsity-promoting dynamic mode decomposition," *Physics of Fluids*, vol. 26, p. 024103, feb 2014.

[124] Kapteyn, M. G., Knezevic, D. J., and Willcox, K., "Toward predictive digital twins via component-based reduced-order models and interpretable machine learning," American Institute of Aeronautics and Astronautics (AIAA), jan 2020.

[125] Karhunen, K., "Zur Spektraltheorie Stochastischer Prozesse," *Ann. Acad. Sci. Fennicae*, no. 37, 1946.

[126] Karlberg, M., Löfstrand, M., Sandberg, S., and Lundin, M., "State of the art in simulation-driven design," *International Journal of Product Development*, vol. 18, no. 1, p. 68, 2013.

[127] Kato, H. and Funazaki, K.-i., "POD-Driven Adaptive Sampling for Efficient Surrogate Modeling and its Application to Supersonic Turbine Optimization," in *Volume 2B: Turbomachinery*, American Society of Mechanical Engineers, jun 2014.

[128] Kennedy, G. J. and Chin, T. W., "A sequential convex optimization method for multimaterial compliance design problems," *Computers & Structures*, vol. 212, pp. 110–124, feb 2019.

[129] Kennedy, M. and O'Hagan, A., "Predicting the output from a complex computer code when fast approximations are available," *Biometrika*, vol. 87, pp. 1–13, mar 2000.

[130] Knyazev, A. V. and Argentati, M. E., "Principal angles between subspaces in an A-based scalar product: Algorithms and perturbation estimates," *SIAM Journal on Scientific Computing*, vol. 23, pp. 2008–2040, jul 2002.

[131] Koizumi, H., Tsutsumi, S., and Haga, T., "Sparsity-Promoting Dynamic Mode Decomposition Analysis on Aeroacoustics of a Clustered Supersonic Jet," in *23rd AIAA Computational Fluid Dynamics Conference*, no. June, (Reston, Virginia), pp. 1–20, American Institute of Aeronautics and Astronautics, jun 2017.

[132] Kokiopoulou, E., Chen, J., and Saad, Y., "Trace optimization and eigenproblems in dimension reduction methods," *Numerical Linear Algebra with Applications*, vol. 18, pp. 565–602, may 2011.

[133] Koopman, B. O., "Hamiltonian Systems and Transformation in Hilbert Space," *Proceedings of the National Academy of Sciences*, vol. 17, pp. 315–318, may 1931.

[134] Krige, D. G., "A statistical approach to some basic mine valuation problems on the Witwatersrand," *Journal of the Southern African Institute of Mining and Metallurgy*, vol. 52, no. 6, pp. 119–139, 1951.

[135] Kutz, J. N., "Deep learning in fluid dynamics," *Journal of Fluid Mechanics*, vol. 814, pp. 1–4, mar 2017.

[136] Kutz, J. N., Brunton, S. L., Brunton, B. W., and Proctor, J. L., *Dynamic Mode Decomposition*. Philadelphia, PA: Society for Industrial and Applied Mathematics, nov 2016.

[137] Kutz, J. N., Brunton, S. L., Luchtenburg, D. M., Rowley, C. W., and Tu, J. H., "On dynamic mode decomposition: Theory and applications," *Journal of Computational Dynamics*, vol. 1, pp. 391–421, dec 2014.

[138] Kutz, J. N., Proctor, J. L., and Brunton, S. L., "Koopman Theory for Partial Differential Equations," pp. 1–21, jul 2016.

[139] Le Clainche, S. and Vega, J. M., "Analyzing Nonlinear Dynamics via Data-Driven Dynamic Mode Decomposition-Like Methods," *Complexity*, vol. 2018, pp. 1–21, dec 2018.

[140] Le Gratiet, L., "Multi-fidelity Gaussian process regression for computer experiments," 2013.

[141] Lee, K., Nam, T., Perullo, C., and Mavris, D. N., "Reduced-Order modeling of a High-Fidelity propulsion system simulation," *AIAA Journal*, vol. 49, pp. 1665–1682, aug 2011.

[142] LeGresley, P. and Alonso, J., "Airfoil design optimization using reduced order models based on proper orthogonal decomposition," in *Fluids 2000 Conference and Exhibit*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, jun 2000.

[143] LeGresley, P. and Alonso, J., "Investigation of non-linear projection for POD based reduced order models for Aerodynamics," in *39th Aerospace Sciences Meeting and Exhibit*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, jan 2001.

[144] Lewis, A., Smith, R., and Williams, B., "Gradient free active subspace construction using Morris screening elementary effects," *Computers & Mathematics with Applications*, vol. 72, pp. 1603–1615, sep 2016.

[145] Li, G., Rosenthal, C., and Rabitz, H., "High dimensional model representations," aug 2001.

[146] Li, J., Cai, J., and Qu, K., "Adjoint-Based Two-Step Optimization Method Using Proper Orthogonal Decomposition and Domain Decomposition," *AIAA Journal*, vol. 56, no. 3, pp. 1133–1145, 2018.

[147] Li, J. and Zhang, W., "The performance of proper orthogonal decomposition in discontinuous flows," *Theoretical and Applied Mechanics Letters*, vol. 6, pp. 236–243, sep 2016.

[148] Li, Q., Dietrich, F., Bollt, E. M., and Kevrekidis, I. G., "Extended dynamic mode decomposition with dictionary learning: A data-driven adaptive spectral decomposition of the Koopman operator," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 27, p. 103111, oct 2017.

[149] Li, S., "Matrix-based decomposition algorithms for engineering applications: the survey and generic framework," *International Journal of Product Development*, vol. 9, no. 1/2/3, pp. 78–110, 2009.

[150] Liberty, E., *Accelerated dense random projections*. No. May, Citeseer, 2009.

[151] Lieberman, C., Willcox, K., and Ghattas, O., "Parameter and State Model Reduction for Large-Scale Statistical Inverse Problems," *SIAM Journal on Scientific Computing*, vol. 32, pp. 2523–2542, jan 2010.

[152] Lieu, T. and Lesoinne, M., "Parameter Adaptation of Reduced Order Models for Three-Dimensional Flutter Analysis," in *42nd AIAA Aerospace Sciences Meeting and Exhibit*, no. January, (Reston, Virigina), pp. 1–9, American Institute of Aeronautics and Astronautics, jan 2004.

[153] Linsenmann, C., Hoppe, R. H., Antil, H., Heinkenschloss, M., and Wixforth, A., "Reduced order modeling based shape optimization of surface acoustic wave driven microfluidic biochips," *Mathematics and Computers in Simulation*, vol. 82, no. 10, pp. 1986–2003, 2010.

[154] Liu, H., Ong, Y.-S., and Cai, J., "A survey of adaptive sampling for global metamodeling in support of simulation-based complex engineering design," *Structural and Multidisciplinary Optimization*, vol. 57, pp. 393–416, jan 2018.

[155] Lorenz, E., "Empirical orthogonal functions and statistical weather prediction," 1956.

[156] Loudon, T. and Pankavich, S., "Mathematical analysis and dynamic active subspaces for a long term model of HIV," *Mathematical Biosciences and Engineering*, vol. 14, pp. 709–733, jun 2017.

[157] Lu, H. and Tartakovsky, D. M., "Predictive Accuracy of Dynamic Mode Decomposition," may 2019.

[158] Lu, K., Jin, Y., Chen, Y., Yang, Y., Hou, L., Zhang, Z., Li, Z., and Fu, C., "Review for order reduction based on proper orthogonal decomposition and outlooks of applications in mechanical systems," *Mechanical Systems and Signal Processing*, vol. 123, pp. 264–297, may 2019.

[159] Lukaczyk, T. W., Constantine, P., Palacios, F., and Alonso, J. J., "Active Subspaces for Shape Optimization," in *10th AIAA Multidisciplinary Design Optimization Conference*, no. January, (Reston, Virginia), pp. 1–18, American Institute of Aeronautics and Astronautics, jan 2014.

[160] Lumley, J. L., "The Structure of Inhomogeneous Turbulence," in *Atmospheric Turbulence and Wave Propagation* (Tatarski, A. M. Y. and I., V., eds.), (Nauka, Moscow), pp. 166–178, 1967.

[161] Ly, H. V. and Tran, H. T., "Modeling and control of physical processes using proper orthogonal decomposition," *Mathematical and Computer Modelling*, vol. 33, pp. 223–236, jan 2001.

[162] Lyu, Z., Kenway, G. K., and Martins, J. R., "Aerodynamic shape optimization investigations of the common research model wing benchmark," *AIAA Journal*, vol. 53, no. 4, pp. 968–985, 2015.

[163] Mainini, L. and Willcox, K., "Surrogate Modeling Approach to Support Real-Time Structural Assessment and Decision Making," *AIAA Journal*, vol. 53, no. 6, pp. 1612–1626, 2015.

[164] Malouin, B., Trépanier, J.-Y., and Gariépy, M., "Interpolation of Transonic Flows Using a Proper Orthogonal Decomposition Method," *International Journal of Aerospace Engineering*, vol. 2013, pp. 1–11, 2013.

[165] Mann, J. and Kutz, J. N., "Dynamic mode decomposition for financial trading strategies," *Quantitative Finance*, vol. 16, pp. 1643–1655, nov 2016.

[166] Martins, J. R. R. A., "Chapter 19: Multidisciplinary Design Optimization of Aerospace Systems," in *Advances and Trends in Optimization with Engineering Applications*, pp. 249–257, Philadelphia, PA: Society for Industrial and Applied Mathematics, apr 2017.

[167] Martins, J. R. R. A. and Lambe, A. B., "Multidisciplinary Design Optimization: A Survey of Architectures," *AIAA Journal*, vol. 51, no. 9, pp. 2049–2075, 2013.

[168] Martinsson, P.-G., "Randomized methods for matrix computations," jul 2016.

[169] Martinsson, P. G. and Voronin, S., "A randomized blocked algorithm for efficiently computing rank-revealing factorizations of matrices," *SIAM Journal on Scientific Computing*, vol. 38, pp. S485–S507, oct 2016.

[170] Matheron, G., "Principles of geostatistics," *Economic Geology*, vol. 58, pp. 1246–1266, dec 1963.

[171] McHutchon, A., "Differentiating gaussian processes," 2013.

[172] Mezić, I., "Spectral Properties of Dynamical Systems, Model Reduction and Decompositions," *Nonlinear Dynamics*, vol. 41, pp. 309–325, aug 2005.

[173] Mezić, I., "Analysis of Fluid Flows via Spectral Properties of the Koopman Operator," *Annual Review of Fluid Mechanics*, vol. 45, pp. 357–378, jan 2013.

[174] Michel Loève, *Probability theory*. Princeton, N. J.-Toronto-New York-London: The University Series in Higher Mathematics. D. Van Nostrand Co., Inc.„ 2nd ed. ed., 1960.

[175] Mifsud, M. J., MacManus, D. G., and Shaw, S. T., "A variable-fidelity aerodynamic model using proper orthogonal decomposition," *International Journal for Numerical Methods in Fluids*, vol. 82, no. 10, pp. 646–663, 2016.

[176] Mihram, G. A., "An Efficient Procedure for Locating the Optimal Simular Response," in *Proceedings of the Fourth Annual Conference on Applications of Simulation*, pp. 154–161, Winter Simulation Conference, 1970.

[177] Myers, A., "Stanford Researchers Break Million-core Supercomputer Barrier," 2013.

[178] Myers, R. H., Montgomery, D. C., and Anderson-Cook, C. M., *Response surface methodology : process and product optimization using designed experiments.* Wiley, 2009.

[179] Nathan Kutz, J., Proctor, J. L., and Brunton, S. L., "Applied Koopman Theory for Partial Differential Equations and Data-Driven Modeling of Spatio-Temporal Systems," *Complexity*, vol. 2018, pp. 1–16, dec 2018.

[180] Negri, F., Manzoni, A., and Amsallem, D., "Efficient model reduction of parametrized systems by matrix discrete empirical interpolation," *Journal of Computational Physics*, vol. 303, pp. 431–454, dec 2015.

[181] Nguyen, N. C., Patera, A. T., and Peraire, J., "A best points' interpolation method for efficient approximation of parametrized functions," *International Journal for Numerical Methods in Engineering*, vol. 73, pp. 521–543, jan 2008.

[182] Noack, B. R., Stankiewicz, W., Morzyński, M., and Schmid, P. J., "Recursive dynamic mode decomposition of transient and post-transient wake flows," *Journal of Fluid Mechanics*, vol. 809, pp. 843–872, dec 2016.

[183] Ohlberger, M. and Rave, S., "Reduced Basis Methods: Success, Limitations and Future Challenges," pp. 1–15, nov 2015.

[184] Ohlberger, M. and Smetana, K., "A Dimensional Reduction Approach Based on the Application of Reduced Basis Methods in the Framework of Hierarchical Model Reduction," *SIAM Journal on Scientific Computing*, vol. 36, pp. A714–A736, jan 2014.

[185] Orszag, S. A. and Patterson, G. S., "Numerical Simulation of Three-Dimensional Homogeneous Isotropic Turbulence," *Physical Review Letters*, vol. 28, pp. 76–79, jan 1972.

[186] Palar, P. S. and Shimoyama, K., "Exploiting active subspaces in global optimization," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion on - GECCO '17*, (New York, New York, USA), pp. 1487–1494, ACM Press, 2017.

[187] Panzer, H., Mohring, J., Eid, R., and Lohmann, B., "Parametric model order reduction by matrix interpolation," *at-Automatisierungstechnik Methoden und Anwendungen der Steuerungs-, Regelungs-und Informationstechnik*, vol. 58, no. 8, pp. 475–484, 2010.

[188] Park, C., Haftka, R. T., and Kim, N. H., "Remarks on multi-fidelity surrogates," *Structural and Multidisciplinary Optimization*, vol. 55, no. 3, pp. 1029–1050, 2017.

[189] Park, K. H., Jun, S. O., Baek, S. M., Cho, M. H., Yee, K. J., and Lee, D. H., "Reduced-Order Model with an Artificial Neural Network for Aerostructural Design Optimization," *Journal of Aircraft*, vol. 50, pp. 1106–1116, jul 2013.

[190] Parrish, J., Rais-Rohani, M., and Janus, M., "Reduced-Order Techniques for Sensitivity Analysis and Design Optimization of Aerospace Systems," *AIAA Journal*, vol. 53, pp. 3567–3576, dec 2015.

[191] Peherstorfer, B. and Willcox, K., "Data-driven operator inference for nonintrusive projection-based model reduction," *Computer Methods in Applied Mechanics and Engineering*, vol. 306, pp. 196–215, 2016.

[192] Peherstorfer, B., Willcox, K., and Gunzburger, M., "Survey of Multifidelity Methods in Uncertainty Propagation, Inference, and Optimization," *SIAM Review*, vol. 60, pp. 550–591, jan 2018.

[193] Per-Gunnar Martinsson, "Randomized Projection Methods in Linear Algebra and Data Analysis," 2018.

[194] Perron, C., *Multi-Fidelity Reduced-Order Modeling applied to Fields with Inconsistent Representations*. PhD thesis, Georgia Institute of Technology, 2020.

[195] Perron, C., Rajaram, D., and Mavris, D., "Development of a Multi-Fidelity Reduced-Order Model Based on Manifold Alignment," in *AIAA AVIATION 2020 FORUM*, (Reston, Virginia), American Institute of Aeronautics and Astronautics, jun 2020.

[196] Pinkus, A., *Ridge Functions*. Cambridge: Cambridge University Press, 2015.

[197] Qian, E., Grepl, M., Veroy, K., and Willcox, K., "A Certified Trust Region Reduced Basis Approach to PDE-Constrained Optimization," *SIAM Journal on Scientific Computing*, vol. 39, pp. S434–S460, jan 2017.

[198] Quan, E., Xu, M., Xie, D., and Li, G., "Efficient Nonlinear Reduced-Order Model for Computational Fluid Dynamics-Based Aeroelastic Analysis," *AIAA Journal*, vol. 56, no. 9, pp. 3701–3717, 2018.

[199] Queipo, N. V., Haftka, R. T., Shyy, W., Goel, T., Vaidyanathan, R., and Kevin Tucker, P., "Surrogate-based analysis and optimization," *Progress in Aerospace Sciences*, vol. 41, pp. 1–28, jan 2005.

[200] Rai, M., "Three-dimensional aerodynamic design using artificial neural networks," in *40th AIAA Aerospace Sciences Meeting & Exhibit*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, jan 2002.

[201] Raissi, M. and Karniadakis, G., "Deep Multi-fidelity Gaussian Processes," apr 2016.

[202] Rajaram, D., Cai, Y., Chakraborty, I., and Mavris, D. N., "Integrated sizing and optimization of aircraft and subsystem architectures in early design," *Journal of Aircraft*, vol. 55, no. 5, pp. 1942–1954, 2018.

[203] Rajaram, D., Gautier, R. H., Perron, C., Pinon-Fischer, O. J., and Mavris, D., "Non-intrusive parametric reduced order models with high-dimensional inputs via gradient-free active subspace," in *AIAA AVIATION 2020 FORUM*, p. 3184, 2020.

[204] Rajaram, D., Perron, C., Puranik, T. G., and Mavris, D. N., "Randomized algorithms for non-intrusive parametric reduced order modeling," *AIAA Journal*, pp. 1–19, 2020.

[205] Rajaram, D., Puranik, T. G., Ashwin Renganathan, S., Sung, W., Fischer, O. P., Mavris, D. N., and Ramamurthy, A., "Empirical Assessment of Deep Gaussian Process Surrogate Models for Engineering Problems," *Journal of Aircraft*, pp. 1–15, sep 2020.

[206] Rajaram, D., Puranik, T. G., Perron, C., and Mavris, D. N., "Non-intrusive parametric reduced order modeling using randomized algorithms," in *AIAA Scitech 2020 Forum*, p. 0417, 2020.

[207] Rajaram, D., Puranik, T. G., Renganathan, S. A., Sung, W., Pinon Fischer, O., Mavris, D. N., and Ramamurthy, A., "Deep gaussian process enabled

surrogate models for aerodynamic flows," in *AIAA Scitech Forum*, American Institute of Aeronautics and Astronautics, 2020.

[208] Rani, M., Dhok, S. B., and Deshmukh, R. B., "A Systematic Review of Compressive Sensing: Concepts, Implementations and Applications," *IEEE Access*, vol. 6, pp. 4875–4894, 2018.

[209] Rasmussen, C. E. and Williams, C. K. I., *Gaussian processes for machine learning*. MIT Press, 2006.

[210] Rathinam, M. and Petzold, L. R., "A New Look at Proper Orthogonal Decomposition," *SIAM Journal on Numerical Analysis*, vol. 41, pp. 1893–1925, jan 2003.

[211] Rayas-Sánchez, J. E., "Power in simplicity with ASM: Tracing the aggressive space mapping algorithm over two decades of development and engineering applications," *IEEE Microwave Magazine*, vol. 17, no. 4, pp. 64–76, 2016.

[212] Raymer, D., *Aircraft Design: A Conceptual Approach, Fifth Edition*. Washington, DC: American Institute of Aeronautics and Astronautics, Inc., aug 2012.

[213] Renganathan, S. A., Liu, Y., and Mavris, D. N., "Koopman-Based Approach to Nonintrusive Projection-Based Reduced-Order Modeling with Black-Box High-Fidelity Models," *AIAA Journal*, vol. 56, pp. 4087–4111, oct 2018.

[214] Ripepi, M., Verveld, M. J., Karcher, N. W., Franz, T., Abu-Zurayk, M., Görtz, S., and Kier, T. M., "Reduced-order models for aerodynamic applications, loads and MDO," *CEAS Aeronautical Journal*, vol. 9, pp. 171–193, mar 2018.

[215] Robinson, T. D., Eldred, M. S., Willcox, K. E., and Haimes, R., "Surrogate-Based Optimization Using Multifidelity Models with Variable Parameterization and Corrected Space Mapping," *AIAA Journal*, vol. 46, no. 11, pp. 2814–2822, 2008.

[216] Rogallo, R. S. and Moin, P., "Numerical Simulation of Turbulent Flows," *Annual Review of Fluid Mechanics*, vol. 16, pp. 99–137, jan 1984.

[217] Rosario, R. D., "Propulsion Technologies for Future Aircraft Generations: Clean, Lean, Quiet, and Green," tech. rep., NASA, 2011.

[218] Roskam, J., *Airplane design*. DARcorporation, 2004.

[219] Rowley, C. W. and Dawson, S. T., "Model Reduction for Flow Analysis and Control," *Annual Review of Fluid Mechanics*, vol. 49, pp. 387–417, jan 2017.

[220] ROWLEY, C. W., MEZIĆ, I., BAGHERI, S., SCHLATTER, P., and HENNINGSON, D. S., "Spectral analysis of nonlinear flows," *Journal of Fluid Mechanics*, vol. 641, p. 115, dec 2009.

[221] Rozza, G., Huynh, D. B. P., and Patera, A. T., "Reduced Basis Approximation and a Posteriori Error Estimation for Affinely Parametrized Elliptic Coercive Partial Differential Equations," *Archives of Computational Methods in Engineering*, vol. 15, pp. 229–275, sep 2008.

[222] Rüde, U., Willcox, K., McInnes, L. C., and Sterck, H. D., "Research and Education in Computational Science and Engineering," *SIAM Review*, vol. 60, pp. 707–754, jan 2018.

[223] Russi, T. M., *Uncertainty Quantification with Experimental Data and Complex System Models*. PhD thesis, University of California Berkeley, 2010.

[224] Sachs, E. W. and Volkwein, S., "POD-galerkin approximations in PDE-constrained optimization," *GAMM Mitteilungen*, vol. 33, no. 2, pp. 194–208, 2010.

[225] Sacks, J., Schiller, S. B., and Welch, W. J., "Designs for Computer Experiments," *Technometrics*, vol. 31, pp. 41–47, feb 1989.

[226] Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P., "Design and analysis of computer experiments," *Statistical science*, pp. 409–423, 1989.

[227] Sarojini, D., Rajaram, D., Solano, D., and Mavris, D. N., "Adjoint-based structural optimization for beam-like structures subjected to dynamic loads," in *AIAA Scitech 2020 Forum*, p. 0273, 2020.

[228] Schaller, R., "Moore's law: past, present and future," *IEEE Spectrum*, vol. 34, pp. 52–59, jun 1997.

[229] Schmid, P. and Sesterhenn, J., "Dynamic Mode Decomposition of numerical and experimental data," *American Physical Society, 61st Annual Meeting of the APS Division of Fluid Dynamics, November 23-25, 2008, abstract id. MR.007*, 2008.

[230] SCHMID, P. J., "Dynamic mode decomposition of numerical and experimental data," *Journal of Fluid Mechanics*, vol. 656, pp. 5–28, aug 2010.

[231] Shan, S. and Wang, G. G., "Metamodeling for High Dimensional Simulation-Based Design Problems," *Journal of Mechanical Design*, vol. 132, p. 051009, may 2010.

[232] Shan, S. and Wang, G. G., "Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions," *Structural and Multidisciplinary Optimization*, vol. 41, pp. 219–241, mar 2010.

[233] Simpson, T., Toropov, V., Balabanov, V., and Viana, F., "Design and Analysis of Computer Experiments in Multidisciplinary Design Optimization: A Review of How Far We Have Come - Or Not," in *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, sep 2008.

[234] Simpson, T., Poplinski, J., Koch, P. N., and Allen, J., "Metamodels for Computer-based Engineering Design: Survey and recommendations," *Engineering with Computers*, vol. 17, pp. 129–150, jul 2001.

[235] Sirovich, L., "Turbulence and the Dynamics of Coherent Structure: I, II and III," *Quarterly Applied Mathematics*, vol. 45, p. 561, 1987.

[236] Skinner, S. and Zare-Behtash, H., "State-of-the-art in aerodynamic shape optimisation methods," *Applied Soft Computing*, vol. 62, pp. 933–962, jan 2018.

[237] Sobieszczanski-Sobieski, J. and Haftka, R. T., "Multidisciplinary aerospace design optimization: survey of recent developments," *Structural Optimization*, vol. 14, pp. 1–23, aug 1997.

[238] Soilahoudine, M., Gogu, C., and Bes, C., "Accelerated Adaptive Surrogate-Based Optimization Through Reduced-Order Modeling," *AIAA Journal*, vol. 55, no. 5, pp. 1681–1694, 2017.

[239] Solano, D., Sarojini, D., Corman, J. A., and Mavris, D. N., "Structural sizing of unconventional aircraft under static and dynamic aeroelastic loading," in *AIAA Scitech 2020 Forum*, p. 0274, 2020.

[240] Sorzano, C. O. S., Vargas, J., and Montano, A. P., "A survey of dimensionality reduction techniques," mar 2014.

[241] Spalart, P. and Allmaras, S., "A one-equation turbulence model for aerodynamic flows," in *30th Aerospace Sciences Meeting and Exhibit*, (Reston, Virigina), American Institute of Aeronautics and Astronautics, jan 1992.

[242] Swischuk, R., Mainini, L., Peherstorfer, B., and Willcox, K., "Projection-based model reduction: Formulations for physics-based machine learning," *Computers & Fluids*, vol. 179, pp. 704–717, jan 2019.

[243] Taira, K., Brunton, S. L., Dawson, S. T. M., Rowley, C. W., Colonius, T., McKeon, B. J., Schmidt, O. T., Gordeyev, S., Theofilis, V., and Ukeiley, L. S., "Modal Analysis of Fluid Flows: An Overview," *AIAA Journal*, vol. 55, pp. 4013–4041, dec 2017.

[244] Tang, C., Gee, K., and Lawrence, S., "Generation of Aerodynamic Data using a Design Of Experiment and Data Fusion Approach," *43rd AIAA Aerospace Sciences Meeting and Exhibit*, no. January, pp. 1–9, 2005.

[245] Tissot, G., Cordier, L., Benard, N., and Noack, B. R., "Model reduction using Dynamic Mode Decomposition," *Comptes Rendus Mécanique*, vol. 342, pp. 410–416, jun 2014.

[246] Toal, D. J. J., "On the potential of a multi-fidelity G-pod based approach for optimization & uncertainty quantification," *Turbine technical conference and exposition*, 2014.

[247] Townsend, J., Koep, N., and Weichwald, S., "Pymanopt: A python toolbox for optimization on manifolds using automatic differentiation," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 4755–4759, 2016.

[248] Trehan, S. and Durlofsky, L. J., "Trajectory piecewise quadratic reduced-order model for subsurface flow, with application to PDE-constrained optimization," *Journal of Computational Physics*, vol. 326, pp. 446–473, 2016.

[249] Tripathy, R., Bilionis, I., and Gonzalez, M., "Gaussian processes with built-in dimensionality reduction: Applications to high-dimensional uncertainty propagation," *Journal of Computational Physics*, vol. 321, pp. 191–223, 2016.

[250] Tripathy, R. K. and Bilionis, I., "Deep uq: Learning deep neural network surrogate models for high dimensional uncertainty quantification," *Journal of computational physics*, vol. 375, pp. 565–588, 2018.

[251] Tropp, J. A., Yurtsever, A., Udell, M., and Cevher, V., "Practical Sketching Algorithms for Low-Rank Matrix Approximation," *SIAM Journal on Matrix Analysis and Applications*, vol. 38, pp. 1454–1485, jan 2017.

[252] Tropp, J. A., Yurtsever, A., Udell, M., and Cevher, V., "Streaming Low-Rank Matrix Approximation with an Application to Scientific Simulation," *SIAM Journal on Scientific Computing*, vol. 41, pp. A2430–A2463, jan 2019.

[253] Ulu, E., Zhang, R., and Kara, L. B., "A data-driven investigation and estimation of optimal topologies under variable loading configurations," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, vol. 4, pp. 61–72, mar 2016.

[254] Van Der Maaten, L., Postma, E., and Van den Herik, J., "Dimensionality Reduction: A Comparative Review," 2009.

[255] Viana, F. A. C., Simpson, T. W., Balabanov, V., and Toropov, V., "Special Section on Multidisciplinary Design Optimization: Metamodeling in Multidisciplinary Design Optimization: How Far Have We Really Come?," *AIAA Journal*, vol. 52, pp. 670–690, apr 2014.

[256] Wang, C. and Mahadevan, S., "A general framework for manifold alignment," in *AAAI Fall Symposium Series*, pp. 53–58, 2009.

[257] Wang, C., Bai, J., and Hesthaven, J. S., "An iterative approach to improve Non-intrusive Reduced-Order Models efficiency for parameterized problems," in *21st AIAA International Space Planes and Hypersonics Technologies Conference*, (Reston, Virginia), American Institute of Aeronautics and Astronautics, mar 2017.

[258] Wang, G. G. and Shan, S., "Review of Metamodeling Techniques in Support of Engineering Design Optimization," *Journal of Mechanical Design*, vol. 129, no. 4, p. 370, 2007.

[259] Wang, X., Kou, J., and Zhang, W., "Multi-fidelity surrogate reduced-order modeling of steady flow estimation," *International Journal for Numerical Methods in Fluids*, no. April, pp. 1–19, 2020.

[260] White, H., *Artificial neural networks: approximation and learning theory*. Blackwell Publishers, Inc., 1992.

[261] Wiart, L., Atinault, O., Grenon, R., Paluch, B., and Hue, D., "Development of NOVA Aircraft Configurations for Large Engine Integration Studies," in *33rd AIAA Applied Aerodynamics Conference*, (Reston, Virginia), American Institute of Aeronautics and Astronautics, jun 2015.

[262] Willcox, K., "Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition," *Computers & fluids*, vol. 35, no. 2, pp. 208–226, 2006.

[263] Williams, M. O., Kevrekidis, I. G., and Rowley, C. W., "A DataDriven Approximation of the Koopman Operator: Extending Dynamic Mode Decomposition," *Journal of Nonlinear Science*, vol. 25, pp. 1307–1346, dec 2015.

[264] Williams, M. O., Schmid, P. J., and Kutz, J. N., "Hybrid Reduced-Order Integration with Proper Orthogonal Decomposition and Dynamic Mode Decomposition," *Multiscale Modeling & Simulation*, vol. 11, pp. 522–544, jan 2013.

[265] Wycoff, N., Binois, M., and Wild, S. M., "Sequential Learning of Active Subspaces," jul 2019.

[266] Wynn, A., Pearson, D. S., Ganapathisubramani, B., and Goulart, P. J., "Optimal mode decomposition for unsteady flows," *Journal of Fluid Mechanics*, vol. 733, pp. 473–503, oct 2013.

[267] Xiao, D., Fang, F., Buchan, A., Pain, C., Navon, I., and Muggeridge, A., "Non-intrusive reduced order modelling of the NavierStokes equations," *Computer Methods in Applied Mechanics and Engineering*, vol. 293, pp. 522–541, aug 2015.

[268] Xiao, D., Fang, F., Pain, C., and Hu, G., "Non-intrusive reduced-order modelling of the Navier-Stokes equations based on RBF interpolation," *International Journal for Numerical Methods in Fluids*, vol. 79, pp. 580–595, dec 2015.

[269] Xiao, D., Yang, P., Fang, F., Xiang, J., Pain, C. C., and Navon, I. M., "Non-intrusive reduced order modelling of fluid-structure interactions," *Computer Methods in Applied Mechanics and Engineering*, vol. 303, pp. 35–54, 2016.

[270] Yeung, P. and Pope, S., "An algorithm for tracking fluid particles in numerical simulations of homogeneous turbulence," *Journal of Computational Physics*, vol. 79, pp. 373–416, dec 1988.

[271] Yin, X. and Li, B., "Sufficient dimension reduction based on an ensemble of minimum average variance estimators," *The Annals of Statistics*, vol. 39, pp. 3392–3416, dec 2011.

[272] Yondo, R., Andrés, E., and Valero, E., "A review on design of experiments and surrogate models in aircraft real-time and many-query aerodynamic analyses," *Progress in Aerospace Sciences*, vol. 96, no. March 2017, pp. 23–61, 2018.

[273] Yu, W., Gu, Y., and Li, Y., "Efficient Randomized Algorithms for the Fixed-Precision Low-Rank Matrix Approximation," *SIAM Journal on Matrix Analysis and Applications*, vol. 39, pp. 1339–1359, jan 2018.

[274] Yue, Y. and Meerbergen, K., "Using Krylov-Padé model order reduction for accelerating design optimization of structures and vibrations in the frequency domain," *International Journal for Numerical Methods in Engineering*, vol. 90, pp. 1207–1232, jun 2012.

[275] Yue, Y. and Meerbergen, K., "Accelerating Optimization of Parametric Linear Systems by Model Order Reduction," *SIAM Journal on Optimization*, vol. 23, pp. 1344–1370, jan 2013.

[276] Yutko, B. M., Titchener, N., Courtin, C., Lieu, M., Wirsing, L., Tylko, J., Jeffrey, C. T., Roberts, T. W., and Church, C. S., "Conceptual Design of a D8 Commercial Aircraft," in *17th AIAA Aviation Technology, Integration, and Operations Conference*, (Reston, Virginia), American Institute of Aeronautics and Astronautics, jun 2017.

[277] Zahm, O., Constantine, P., Prieur, C., and Marzouk, Y., "Gradient-based dimension reduction of multivariate vector-valued functions," jan 2018.

[278] Zahr, M. J., Avery, P., and Farhat, C., "A multilevel projection-based model order reduction framework for nonlinear dynamic multiscale problems in structural and solid mechanics," *International Journal for Numerical Methods in Engineering*, vol. 112, pp. 855–881, nov 2017.

[279] Zahr, M. J. and Farhat, C., "Progressive construction of a parametric reduced-order model for PDE-constrained optimization," *International Journal for Numerical Methods in Engineering*, vol. 102, pp. 1111–1135, may 2015.

[280] Zhang, W. and Wei, M., "Model order reduction using DMD modes and adjoint DMD modes," in *8th AIAA Theoretical Fluid Mechanics Conference*, no. June, (Reston, Virginia), pp. 1–10, American Institute of Aeronautics and Astronautics, jun 2017.

[281] Zhang, Y., Sung, W. J., and Mavris, D. N., "Application of Convolutional Neural Network to Predict Airfoil Lift Coefficient," in *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, (Reston, Virginia), American Institute of Aeronautics and Astronautics, jan 2018.

[282] Zimmermann, R. and Görtz, S., "Improved extrapolation of steady turbulent aerodynamics using a non-linear POD-based reduced order model," *The Aeronautical Journal*, vol. 116, pp. 1079–1100, oct 2012.

[283] Zimmermann, R. and Willcox, K., "An Accelerated Greedy Missing Point Estimation Procedure," *SIAM Journal on Scientific Computing*, vol. 38, pp. A2827–A2850, jan 2016.

[284] Zimmermann, R., "Manifold interpolation and model reduction," pp. 1–36, feb 2019.

[285] Zimmermann, R. and Görtz, S., "Non-linear reduced order models for steady aerodynamics," *Procedia Computer Science*, vol. 1, pp. 165–174, may 2010.

[286] Zimmermann, R., Vendl, A., and Görtz, S., "Reduced-Order Modeling of Steady Flows Subject to Aerodynamic Constraints," *AIAA Journal*, vol. 52, pp. 255–266, feb 2014.